NBS
PUBLICATIONS

A11106 978219

NBS SPECIAL PUBLICATION 400-77

U.S. DEPARTMENT OF COMMERCE/National Bureau of Standards

Semiconductor Measurement Technology:

MOS1: A Program for Two-Dimensional Analysis of Si MOSFETs

Charles L. Wilson and James L. Blue

QC
100
.U57
#400-77
1985

The National Bureau of Standards[1] was established by an act of Congress on March 3, 1901. The Bureau's overall goal is to strengthen and advance the nation's science and technology and facilitate their effective application for public benefit. To this end, the Bureau conducts research and provides: (1) a basis for the nation's physical measurement system, (2) scientific and technological services for industry and government, (3) a technical basis for equity in trade, and (4) technical services to promote public safety. The Bureau's technical work is performed by the National Measurement Laboratory, the National Engineering Laboratory, the Institute for Computer Sciences and Technology, and the Center for Materials Science.

The National Measurement Laboratory

Provides the national system of physical and chemical measurement; coordinates the system with measurement systems of other nations and furnishes essential services leading to accurate and uniform physical and chemical measurement throughout the Nation's scientific community, industry, and commerce; provides advisory and research services to other Government agencies; conducts physical and chemical research; develops, produces, and distributes Standard Reference Materials; and provides calibration services. The Laboratory consists of the following centers:

- Basic Standards[2]
- Radiation Research
- Chemical Physics
- Analytical Chemistry

The National Engineering Laboratory

Provides technology and technical services to the public and private sectors to address national needs and to solve national problems; conducts research in engineering and applied science in support of these efforts; builds and maintains competence in the necessary disciplines required to carry out this research and technical service; develops engineering data and measurement capabilities; provides engineering measurement traceability services; develops test methods and proposes engineering standards and code changes; develops and proposes new engineering practices; and develops and improves mechanisms to transfer results of its research to the ultimate user. The Laboratory consists of the following centers:

- Applied Mathematics
- Electronics and Electrical Engineering[2]
- Manufacturing Engineering
- Building Technology
- Fire Research
- Chemical Engineering[2]

The Institute for Computer Sciences and Technology

Conducts research and provides scientific and technical services to aid Federal agencies in the selection, acquisition, application, and use of computer technology to improve effectiveness and economy in Government operations in accordance with Public Law 89-306 (40 U.S.C. 759), relevant Executive Orders, and other directives; carries out this mission by managing the Federal Information Processing Standards Program, developing Federal ADP standards guidelines, and managing Federal participation in ADP voluntary standardization activities; provides scientific and technological advisory services and assistance to Federal agencies; and provides the technical foundation for computer-related policies of the Federal Government. The Institute consists of the following centers:

- Programming Science and Technology
- Computer Systems Engineering

The Center for Materials Science

Conducts research and provides measurements, data, standards, reference materials, quantitative understanding and other technical information fundamental to the processing, structure, properties and performance of materials; addresses the scientific basis for new advanced materials technologies; plans research around cross-country scientific themes such as nondestructive evaluation and phase diagram development; oversees Bureau-wide technical programs in nuclear reactor radiation research and nondestructive evaluation; and broadly disseminates generic technical information resulting from its programs. The Center consists of the following Divisions:

- Inorganic Materials
- Fracture and Deformation[3]
- Polymers
- Metallurgy
- Reactor Radiation

[1]Headquarters and Laboratories at Gaithersburg, MD, unless otherwise noted; mailing address Gaithersburg, MD 20899.
[2]Some divisions within the center are located at Boulder, CO 80303.
[3]Located at Boulder, CO, with some elements at Gaithersburg, MD.

NBS
RESEARCH
INFORMATION
CENTER

Semiconductor Measurement Technology:

MOS1: A Program for Two-Dimensional Analysis of Si MOSFETs

Charles L. Wilson
Semiconductor Devices and Circuits Division
Center for Electronics and Electrical Engineering

and

James L. Blue
Scientific Computing Division
Center for Applied Mathematics

National Engineering Laboratory
National Bureau of Standards
Gaithersburg, MD 20899

U.S. DEPARTMENT OF COMMERCE, Malcolm Baldrige, Secretary
NATIONAL BUREAU OF STANDARDS, Ernest Ambler, Director

Issued April 1985

Library of Congress Catalog Card Number: 85-600520

National Bureau of Standards Special Publication 400-77
Natl. Bur. Stand. (U.S.), Spec. Publ. 400-77, 61 pages (Apr. 1985)
CODEN: XNBSAV

U.S. GOVERNMENT PRINTING OFFICE
WASHINGTON: 1985

For sale by the Superintendent of Documents, U.S. Government Printing Office, Washington, DC 20402

Table of Contents

Page

Abstract 1

1. Introduction 1

2. Model Formulation 2

2.1 Physical Formulation of Semiconductor Device Problems 2

2.2 Boundary Conditions 7

2.3 Doping Profiles 9

3. Numerical Procedures 10

4. Interactive Solution Strategies 14

5. Program Specifics 20

6. Examples 21

6.1 Modeling of Phosphorus S-D Device 21

6.2 Modeling of Arsenic Device 35

7. Summary 35

References 38

Appendix A–File MOS1.DOC 40

Appendix B–File MOS1.EXP 52

List of Figures

Page

1. The coordinate system used in these calculations is shown, imposed on a typical MOS transistor 5
2. Typical values of the field dependence of mobility used in these calculations . . 6
3. Division of a triangle into four similar triangles during mesh refinement 13
4. Primitive mesh for MOS transistor 15
5. Level 1 mesh for MOS transistor 17
6. Initial guess for MOS transistor 18
7. Final potential for MOS transistor 19
8. The measured subthreshold characteristic of the 8.17 – μm transistor 23
9. The measured output characteristics of the 8.17 – μm transistor 24
10. The potential, electron density, and hole density in the 8.17 – μm transistor at a bias of $V_g = 2.0$ V and $V_d = 0.25$ V 25
11. The potential, electron density, and hole density in the 8.17 – μm transistor at a bias of $V_g = 2.0$ V and $V_d = 2.0$ V 26
12. The measured subthreshold characteristic of the 1.83 – μm transistor 27
13. The measured output characteristics of the 1.83 – μm transistor 28
14. The potential, electron density, and hole density in the 1.83 – μm transistor at a bias of $V_g = 2.0$ V and $V_d = 0.25$ V 29
15. The potential, electron density, and hole density in the 1.83 – μm transistor at a bias of $V_g = 2.0$ V and $V_d = 2.0$ V 30
16. The measured output characteristics of the 0.80 – μm transistor 31
17. The potential, electron density, and hole density in the 0.80 – μm transistor at a bias of $V_g = 2.0$ V and $V_d = 0.25$ V 32
18. The potential, electron density, and hole density in the 0.80 – μm transistor at a bias of $V_g = 2.0$ V and $V_d = 2.0$ V 33
19. The potential, electron density, and hole density in the 0.80 – μm transistor at a bias of $V_g = 2.0$ V and $V_d = 3.0$ V 34
20. The measured subthreshold characteristic of the 2.12 – μm transistor 36
21. The measured output characteristics of the 2.12 – μm transistor 37

Semiconductor Measurement Technology:

MOS1:
A Program for Two-Dimensional Analysis of Si MOSFETs

Charles L. Wilson
Semiconductor Devices and Circuits Division
Center for Electronics and Electrical Engineering

and

James L. Blue
Scientific Computing Division
Center for Applied Mathematics
National Bureau of Standards
Gaithersburg, MD 20899

Abstract

The MOS1 program is a portable FORTRAN 77 program suitable for analysis of currents and fields in VLSI devices. The program solves three coupled nonlinear elliptic partial differential semiconductor device equations in two dimensions. Historically, these equations have been solved using a special-purpose program and batch runs on a large, fast computer. We use a general-purpose program which runs on a large minicomputer or scientific workstation. This report discusses the physical formulation of the semiconductor equations and the methods used to select the solution strategy.

Key Words: finite elements; MOS transistor; MOSFET models; partial differential equations; semiconductor device model; simulation.

1. Introduction

Analyzing currents and fields in VLSI devices requires solving three coupled nonlinear elliptic partial differential equations in two dimensions. Historically, these equations have been solved using a special-purpose program and batch runs on a large, fast computer. We use a general-purpose program and interactive runs on a large minicomputer or scientific workstation.

We discuss the physical formulation of the semiconductor equations used in the program, and give example solutions for a short-channel MOSFET in several bias regions. This MOSFET is typical of present-day short-channel VLSI devices.

In section 2, the formulation of the semiconductor device equations used in the model is presented. In section 3, the numerical procedures used in the general-purpose partial differential equation solver, B2DE, are discussed. In section 4, the methods used to select the built-in solution strategy are discussed. In section 5, specific program implementation details are given. In section 6, the example results are presented.

2. Model Formulation

2.1 Physical Formulation of Semiconductor Device Problems

The physical formulation of the device simulations discussed here requires solution of the basic semiconductor device equations, using the appropriate carrier distribution functions, Boltzmann or Fermi-Dirac, with appropriate boundary conditions. In addition, the interior structure of each device requires the specification of a two-dimensional doping profile.

The model includes a two-dimensional calculation of the electrostatic potential in the semiconductor and in the oxide and a two-dimensional calculation of the hole and electron quasi-Fermi levels in the semiconductor. With an oxide present, these equations are more general than B2DE can solve directly, since B2DE assumes that all partial differential equations (PDEs) are defined on the same region [1]. We use a fast Poisson solver in planar oxides.

The basic formulation of the model includes a two-dimensional calculation of the electrostatic potential in the semiconductor and in the oxide and a two-dimensional calculation of the hole and electron quasi-Fermi levels and Shockley-Read-Hall recombination in the bulk silicon. The model is based on the basic semiconductor equations:

$$\nabla\cdot(\kappa\,\nabla\,\psi) = -\frac{q}{\epsilon_0}(p - n + N) \tag{1}$$

$$\nabla\cdot(\mu_n\, n\,\nabla\,\phi_n) = \frac{\partial n}{\partial t} + R \tag{2}$$

$$\nabla\cdot(\mu_p\, p\,\nabla\,\phi_p) = \frac{\partial p}{\partial t} - R \tag{3}$$

where the electron density, n, and the hole density, p, are given for Boltzmann statistics by:

$$n = n_i \exp(q(\psi - \phi_n)/kT) \tag{4}$$

$$p = n_i \exp(q(\phi_p - \psi)/kT) \tag{5}$$

or for Fermi-Dirac statistics by:

$$n = N_c\, \mathcal{F}_{\frac{1}{2}}(q(\psi - \phi_n - E_g/2 - \frac{kT}{2q}\ln(\frac{N_c}{N_v}))/kT) \tag{6}$$

$$p = N_v\, \mathcal{F}_{\frac{1}{2}}(q(\phi_p - \psi - E_g/2 + \frac{kT}{2q}\ln(\frac{N_c}{N_v}))/kT) \tag{7}$$

where:

$$\mathcal{F}_j(\eta) = \frac{1}{\Gamma(j+1)}\int_0^\infty \frac{\epsilon^j\, d\epsilon}{1 + \exp(\epsilon - \eta)} \tag{8}$$

and where Shockley-Read-Hall recombination is given by:

$$R = \frac{(pn - n_i^2)}{\tau_{p0}(n + n_1) + \tau_{n0}(p + p_1)}. \tag{9}$$

The symbols have their usual meaning as defined in table 1 and are derived from the basic equation given in standard texts such as [2].

Table 1
Symbols

κ	dielectric constant
ψ	electrostatic potential
q	electronic charge
ϵ_0	permittivity of free space
p	hole density
n	electron density
N	net ionized impurity density
μ_n	electron mobility
ϕ_n	electron quasi-Fermi level
R	recombination-generation rate
μ_p	hole mobility
ϕ_p	hole quasi-Fermi level
N_c	density of states in the conduction band
E_g	bandgap
T	Kelvin temperature
N_v	density of states in the valence band
n_i	intrinsic carrier concentration
τ_{n0}	electron lifetime
n_1	electron density at trap energy
τ_{p0}	hole lifetime
p_1	hole density at trap energy
V	applied voltage
$\mathbf{n}$	unit normal to boundary
Q_{total}	total interface charge in the model
Q_{it}	interface trapped charge
Q_{ox}	oxide trapped charge
ϕ_S	surface potential
E_d	energy of donor-like trap
E_a	energy of acceptor-like trap
g_d	donor state spin degeneracy
g_a	acceptor state spin degeneracy
s_n	electron surface recombination velocity
s_p	hole surface recombination velocity
n_0	equilibrium electron density
p_0	equilibrium hole density

A simplification to three static two-dimensional equations is achieved by assuming that bias levels in the device are maintained long enough for static processes to determine the solution; this eliminates time derivatives in eqs (2) and (3). Equations (4) and (5), or (6), (7), and (8), and (9) are substituted into eqs (1),(2), and (3) to yield a static model of the device which includes the effects of steady-state recombination and may include the effect

of the Fermi-Dirac integral function (eq (6)) or of Boltzmann statistics given by eqs (4) and (5). The coordinate system used in these calculations is shown, imposed on a typical MOS transistor, in figure 1.

Doping and mobilities are specified by the user parameters which control the appropriate term generation subroutines.

The validity of eq (1) is limited by the validity of characterizing the polarizability by a simple dielectric constant. This is not a significant limitation in most applications in silicon. Using a tensor instead of a constant would require a fairly simple change in B2DE.

The restriction imposed by the formulation of eqs (2) and (3) is more basic. Each of these equations depends on the existence of an isotropic mobility and on the correctness of a distribution function relating carrier densities to electrostatic and quasi-Fermi potentials. In most applications, a mobility can be defined which meets the above requirements [3]. Anisotropic mobilities could be used with the altered form of B2DE mentioned in the previous paragraph.

High field effects are included in MOS1 using a simple mobility model. The mobility in the channel is described by an empirical formula that was chosen to fit the form of the data given in [4]. A value of low field mobility is needed for each device technology and is obtained from long-channel transistor measurements using the charge-sheet equations given in [5] and the dimensional and doping parameters used in the two-dimensional model. This mobility is then included in the mobility model given by:

$E_y < E_{knee}$

$$\mu = \mu_1 = \mu_0/(1 + k_{E_x} E_x) \tag{10}$$

$E_{knee} < E_y < E_{max}$

$$\mu = \mu_1/\mu_0(v_{max} - b_\mu(E_y - E_{max})^2)/E_y \tag{11}$$

$E_{max} < E_y$

$$\mu = v_{max}/E_y, \tag{12}$$

where E_y is the component of the electric field perpendicular to the channel, E_x is the component of the electric field parallel to the channel, E_{knee} is the field at the lower end of the field dependent region of the model, and E_{max} is the field at the saturated velocity, v_{max}. The parameters E_{knee}, E_{max}, v_{max}, μ_0, k_{E_x}, and b_μ are chosen to fit the data in [4]. Typical values of the field dependence of mobility are shown in figure 2. These parameters have been found to be processing and device independent when tested on samples from five different sources provided that the low field mobility is obtained as described above. This model is implemented in the subroutines MUN and MUP; any similar model could be used by altering these routines.

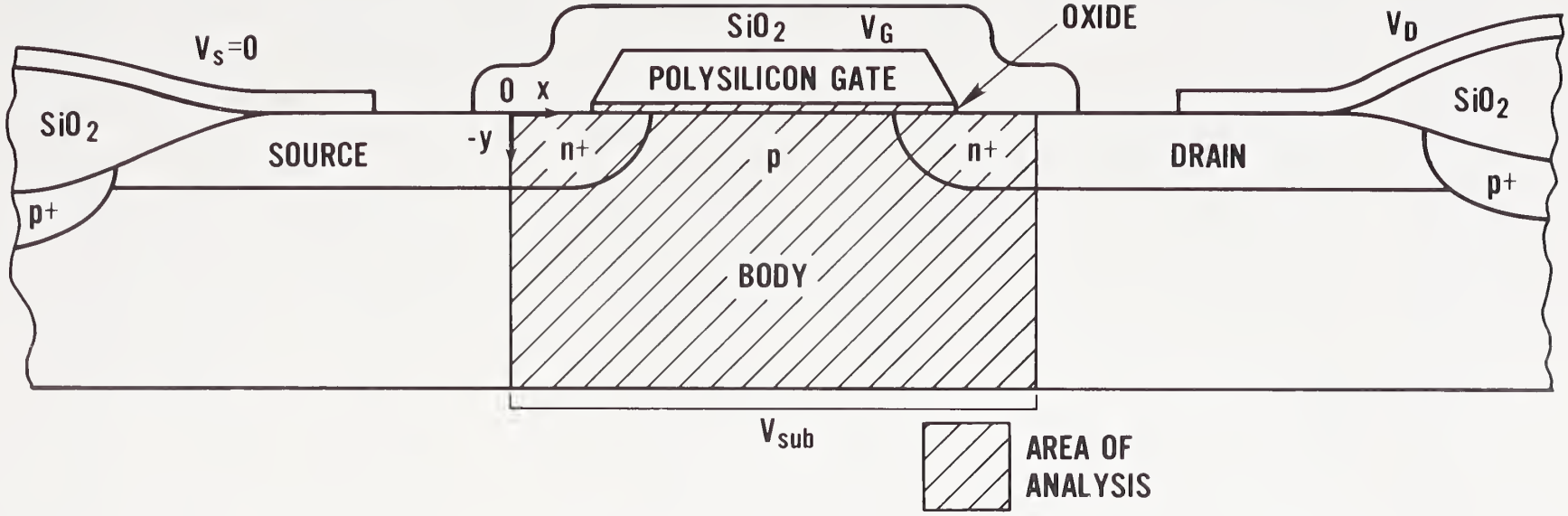

Figure 1. The coordinate system used in these calculations is shown, imposed on a typical MOS transistor.

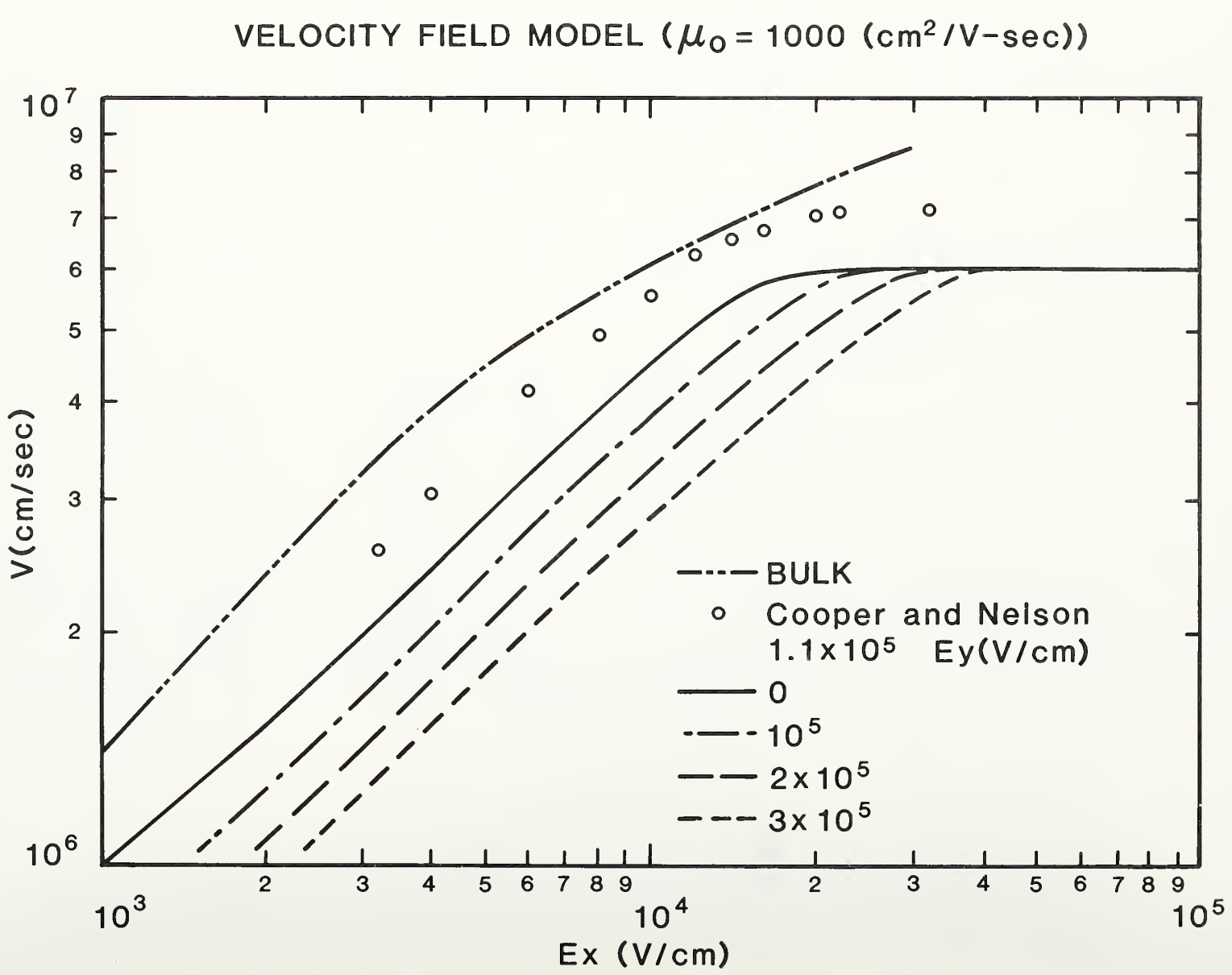

Figure 2. Typical values of the field dependence of mobility used in these calculations.

These mobilities are correct for the channel, the only place where appreciable current flows. In the rest of the transistor, the mobility is a function of the impurity density only. The functional form of this dependence is based on empirical formulas that approximate the data in [2].

In each of the triangular finite elements, the carrier densities used in eqs (2) and (3) are approximated by one of the distribution functions. The factor multiplying the gradient is an exponential function of the solution variables. In the finite-element solution in B2DE, integrals over triangles are approximated by a four-point quadrature rule, using the triangle vertices and the midpoint. The finite-element solution maintains the continuity of the currents only approximately, since the integrals are not done exactly. In high-current regions of the device, the mesh needs to be refined to improve current continuity.

The recombination term given in eq (9) is the usual Shockley-Read-Hall term. For the MOS transistors considered here, where recombination takes place in the lightly doped substrate, this term is usually sufficient. The accuracy of this model is dependent on the models used for τ_{n0} and τ_{p0} as functions of position and doping.

The validity of the distribution functions is more suspect. Boltzmann statistics are only an approximation to Fermi-Dirac statistics. When the latter are used, the distribution functions for holes and electrons given in eqs (6) and (7) assume parabolic densities of states in the valence and conduction bands. This may not always be sufficient.

Including Fermi statistics improves the stability of the solution process by bounding the exponential terms found in Boltzmann statistics. This is significant only when the carrier densities become large, approaching the density of states.

This simplified model neglects carrier interactions and other effects which would affect carrier statistics in a strongly inverted device. In addition, all heavy doping effects which would affect the density of states in the band have been excluded.

B2DE is adequate to handle most of this more complicated physics. It is harder to decide what physics to use than to fit it into the model.

2.2 Boundary Conditions

Most previous models of semiconductor devices have used idealized metallic contacts and idealized oxide-semiconductor interfaces. In the models developed here, more realistic boundary conditions can be used, including both ideal ohmic contacts and contacts with Schottky barriers. For Boltzmann statistics, contacts with an applied potential V are characterized by:

$$\psi = V + \ln(N/n_i) + \phi_{MS} \tag{13}$$

$$\phi_n = V + \phi_{MS} \tag{14}$$

$$\phi_p = V + \phi_{MS}. \tag{15}$$

ϕ_{MS} is zero for a perfect ohmic material. These functions cause the right-hand side of eq (1) to be zero at the metal-semiconductor interface.

In the case of Fermi statistics on n-type material:

$$\psi = V + \mathcal{F}_{\frac{1}{2}}^{-1}(N/N_c) + E_g/2 + \frac{kT}{2q}\ln(\frac{N_c}{N_v}) + \phi_{MS} \tag{16}$$

$$\phi_n = V + \phi_{MS} \tag{17}$$

$$\phi_p = V + \phi_{MS}. \tag{18}$$

On p-type material, eq (16) becomes:

$$\psi = V + \mathcal{F}_{\frac{1}{2}}^{-1}(-N/N_v) - E_g/2 + \frac{kT}{2q}\ln(\frac{N_c}{N_v}) + \phi_{MS}. \tag{19}$$

These functions also cause the right-hand side of eq (1) to be zero at the metal-semiconductor interface.

Any of these sets of boundary conditions are also equivalent to the case of an infinite surface recombination velocity in that the value of the excess carrier density goes to zero. The model of the Schottky contact shown in eqs (13-19) is a simple metal-semiconductor work function difference, ϕ_{MS}. Inclusion of a boundary of this type requires, or will cause, additional mesh refinement in the region of the Schottky contact.

The oxide boundary conditions used are:

$$\frac{\partial \psi_{Si}}{\partial \mathbf{n}} = -\frac{\kappa_{ox}}{\kappa_{Si}}\frac{\partial \psi_{ox}}{\partial \mathbf{n}} - \frac{Q_{total}}{\kappa_{Si}\epsilon_0} \tag{20}$$

$$\frac{\partial \phi_n}{\partial \mathbf{n}} = 0 \tag{21}$$

$$\frac{\partial \phi_p}{\partial \mathbf{n}} = 0. \tag{22}$$

In this formulation of the problem, the interface charge is given by:

$$Q_{total} = Q_{it} + Q_{ox}, \tag{23}$$

where:

$$Q_{it}(\phi_S) = \int_{E_v}^{E_c} \frac{Q(E_d)dE_d}{1 + g_d \exp((\phi_S - E_d)/kT)} - \int_{E_v}^{E_c} \frac{Q(E_a)dE_a}{1 + g_a \exp((E_a - \phi_S)/kT)}. \tag{24}$$

In the present model, the value of the field in the oxide is calculated outside the Newton iteration loop in B2DE. The calculation for planar oxides is done using a fast Poisson solver [6]. The change in potential perpendicular to the surface is large compared to the change parallel to the surface in MOS transistors; the oxide is "thin". This makes it necessary to have a relatively dense mesh in the oxide parallel to the interface, but allows a relatively coarse mesh perpendicular to the interface. In many applications, it is possible to approximate the field in a "thin" oxide by dividing the potential difference across the oxide by the oxide thickness. This approximation is useful because the interface portion of the Jacobian matrix can be obtained exactly.

Symmetry lines are often introduced to bound the device being simulated. These false symmetry lines have boundary conditions identical to real symmetry lines, but are used as boundaries only to limit the computational region of a large device, saving mesh points and reducing calculation cost. The intersection of p-n junctions with these boundaries can cause accuracy and convergence problems. If the false lines of symmetry are too close to the active region of the device, accuracy problems are caused; field distortion propagates into the electrically active region of the device. This also reduces convergence rates. The mesh must be refined in such regions.

2.3 Doping Profiles

The doping profiles used in these calculations enter into the equations only through $N(x, y)$, and possibly some parameters in R, such as τ_{n0} and τ_{p0}. The functions used are not attached to the mesh in any way. This allows the doping to be altered during the calculation. The doping profiles are generated in the subroutine DOPING; the profiles may be altered by modifying this subroutine. The present version of the program only allows the profile types described below.

The most commonly used two-dimensional doping profiles result from the redistribution of impurities by diffusion [7,8,9]. The simplest form of two-dimensional profile is the profile [7] combining a Gaussian profile perpendicular to the surface with an error function complement under the mask edge. This profile is a reasonable representation of the redistribution in ion-implanted impurities at low dose and high annealing temperatures. More abrupt two-dimensional profiles result at high dose and at lower annealing temperatures. These profiles have been studied in two dimensions [8] using the concentration-dependent diffusion model of Fair [10]. Profiles of this type can be modeled by generating two-dimensional profiles of the constant coefficient type with concentration-dependent diffusion lengths.

Low annealing temperatures generate impurity profiles which are not characteristic of diffusion processes. Either these profiles show little or no impurity redistribution [11], or they are the result of impurity redistribution processes in which the ion implantation damage has a significant effect [12]. Profiles of this type are well approximated by a Gaussian under the mask window and a Gaussian rotated about the median range point under the mask.

The channel profile is given by a table of y values measured down from the SiO_2 interface. The profile is constant along the x direction. The end points of the profile are used for values of the y coordinate outside the specified range.

3. Numerical Procedures

An important feature of this work is that the numerical solutions were obtained using general-purpose software for solving elliptic partial differential equations. Although some efficiency is sacrificed, the benefits obtained by using the general-purpose software far outweigh the loss in efficiency. Problems with exact solutions have been solved to test the correctness and accuracy of the codes and to determine optimal strategies and optimal values for parameters in the code. This is essential in separating physical errors from numerical errors and programming errors.

The numerical software was inspired by finite-element software of Bank and Sherman [13, 14] and retains most of their philosophy. The current version of the package solves systems of N PDEs, each of the form:

$$\nabla \cdot [a_i(x, y, \mathbf{u}, \partial\mathbf{u}/\partial x, \partial\mathbf{u}/\partial y)\nabla u_i] = f_i(x, y, \mathbf{u}, \partial\mathbf{u}/\partial x, \partial\mathbf{u}/\partial y) \quad i = 1, 2, \ldots, N \tag{25}$$

in a region bounded by straight line segments and circular arc segments. On each segment of the boundary, each of the u_i must obey either a nonlinear Dirichlet boundary condition type 1:

$$g_i(x, y, \mathbf{u}) = 0 \tag{26}$$

or a nonlinear normal derivative boundary condition type 2:

$$\frac{\partial u_i}{\partial \mathbf{n}} + g_i(x, y, \mathbf{u}) = 0. \tag{27}$$

On a given boundary segment, the type of boundary condition may be different for different PDEs. This is a fairly general example of a system of elliptic PDEs arising from conservation of a flux.

A module which solves a system of linearized elliptic PDEs using linear finite elements on a mesh of triangles is at the heart of the package. (Boundary "triangles" may have one curved side.) The initial triangulation is an input to the package; succeeding triangulations are calculated adaptively by the package. For a given triangulation with M vertices, the calculation proceeds as follows.

The M finite-element basis functions $\{b_m\}$ are linear on each triangle; b_m is 1 at vertex m and 0 at all other vertices. For the ith PDE, the set of vertices on type 1 boundaries is denoted by D_i (for Dirichlet). The solution is approximated by a sum of basis functions

$$u_i(x, y) = \sum_{m=1}^{M} \alpha_{im} b_m(x, y). \tag{28}$$

The coefficients are determined by a Galerkin method [15]; the error in the ith PDE is made orthogonal to each of the basis functions except those in D_i:

$$\int\int [-\nabla \cdot (a_i \nabla u_i) + f]\, b_m = 0, \quad m = 1, 2, \ldots, M, m \text{ not in } D_i \tag{29}$$

The remaining conditions on the α's are that the type 1 boundary conditions hold exactly [15]:

$$g_i\,(x_m, y_m, \mathbf{u}(x_m, y_m)) = 0, \quad m \text{ in } D_i. \tag{30}$$

Equation (29) is integrated by parts to yield

$$\int\int (a_i \nabla u_i \cdot \nabla b_m + f b_m) + \int_2 a_i g_i b_m = 0, \tag{31}$$

where the single integral is over the type-2 parts of the boundary only. Equations (30) and (31) comprise MN nonlinear equations in MN unknowns, the coefficients $\{\alpha_{im}\}$.

The MN nonlinear equations are solved by an iterative process, a damped Newton's method. Let $\mathbf{v}$ be the Newton correction to $\mathbf{u}$; v_i has the same form as u_i,

$$v_i(x, y) = \sum_{m=1}^{M} \beta_{im} b_m(x, y). \tag{32}$$

To calculate $\mathbf{v}$, replace u_i by $u_i + v_i$ in eqs (30) and (31) and linearize in v_i. The linearization of eq (30) yields

$$g_i\,(x_m, y_m, \mathbf{u}(x_m, y_m)) + \sum_{j=1}^{N} \frac{\partial g_i\,(x_m, y_m, \mathbf{u}(x_m, y_m))}{\partial u_j} \beta_{jm} = 0, \quad m \text{ in } D_i, \tag{33}$$

and the linearization of eq (31) yields

$$\int\int \Bigg\{ a_i \nabla v_i \cdot \nabla b_m +$$

$$\nabla u_i \cdot \nabla b_m \sum_{j=1}^{N} \left[\frac{\partial a_i}{\partial u_j} v_j + \frac{\partial a_i}{\partial (\partial u_j/\partial x)} \frac{\partial v_j}{\partial x} + \frac{\partial a_i}{\partial (\partial u_j/\partial y)} \frac{\partial v_j}{\partial x} \right]$$

$$+ b_m \sum_{j=1}^{N} \left[\frac{\partial f_i}{\partial u_j} v_j + \frac{\partial f_i}{\partial (\partial u_j/\partial x)} \frac{\partial v_j}{\partial x} + \frac{\partial f_i}{\partial (\partial u_j/\partial x)} \frac{\partial v_j}{\partial y} \right] \Bigg\}$$

$$+ \int_2 b_m \sum_{j=1}^{N} \left[a_i \frac{\partial g_i}{\partial u_j} v_j + g_i \frac{\partial a_i}{\partial u_j} v_j + g_i \frac{\partial a_i}{\partial (\partial u_j/\partial x)} \frac{\partial v_j}{\partial x} + g_i \frac{\partial a_i}{\partial (\partial u_j/\partial y)} \frac{\partial v_j}{\partial y} \right]$$

$$= -\int\int (a_i \nabla u_i \cdot \nabla b_m + f_i b_m) - \int_2 a_i g_i b_m \equiv r_{im}. \tag{34}$$

The right-hand side is the residual, $r_{im}(\mathbf{u})$. Substituting for $\mathbf{u}$ and $\mathbf{v}$ from eqs (28) and (32) and doing the integrals numerically give a set of MN linear equations for the correction coefficients $\{\beta_{im}\}$. These linear equations are sparse; there are typically about $9N$ nonzero elements per row, on average, and MN can be a few hundred or a few thousand. On the initial triangulation, with MN a few hundred, a satisfactory way to solve these equations is directly, with sparse Gaussian elimination [16]. On later triangulations, with MN up to a few thousand, direct solutions take too much time and space; iterative methods are usually better, especially multilevel iterative methods.

Unless $\mathbf{u}$ is close to a solution of the nonlinear finite-element equations, $\mathbf{u} + \mathbf{v}$ may be worse than $\mathbf{u}$. A damped Newton's method is used: replace $\mathbf{u}$ by $\mathbf{u} + \lambda\,\mathbf{v}$, where λ is chosen so that $\| \mathbf{r}(\mathbf{u} + \lambda\mathbf{v}) \| < \| \mathbf{r}(\mathbf{u}) \|$ where $\| \mathbf{r} \|$ is the Euclidean norm of $\mathbf{r}$. This avoids divergence of the iterative process, but does not guarantee convergence.

In practice, the initial triangulation is usually too coarse to give the desired accuracy; local refinement is needed to represent the solution accurately in part or all of the region. After the package has converged to an approximate solution on the initial triangulation, the approximate solution may be used to estimate the error and produce a new and finer triangulation adaptively. There are various ways to estimate the error in each triangle. For the present work, a local method is used.

Consider a single triangle; for simplicity we suppose its vertices are 1, 2, and 3. The approximate solution inside the triangle is

$$u_i(x, y) = \sum_{m=1}^{3} \alpha_{im} b_m(x, y). \tag{35}$$

We tentatively divide the triangle into four similar triangles (fig. 3). On the four triangles, a potentially more accurate u_i is

$$u'_i(x, y) = \sum_{m=1}^{6} \alpha'_{im} b'_m(x, y). \tag{36}$$

(The primes refer to the finer triangulation.) We keep the old values at the original 3 vertices; $\alpha'_{im} = \alpha_{im}$, $m = 1, 2, 3$. Values at the 3 new vertices are found by solving the linearized finite element equations for α'_{im}, $m = 4, 5, 6$. The error in the triangle can be estimated by the largest change at any of the points 4, 5, and 6.

This estimate is not accurate in absolute value, but the relative estimates in different triangles are sufficiently accurate for refining the triangulation.

The refined triangulation is obtained by dividing the triangles with the largest estimated errors, such as all triangles with estimated error larger than 1/4 of the largest error, or the worst 1/4 of the triangles.

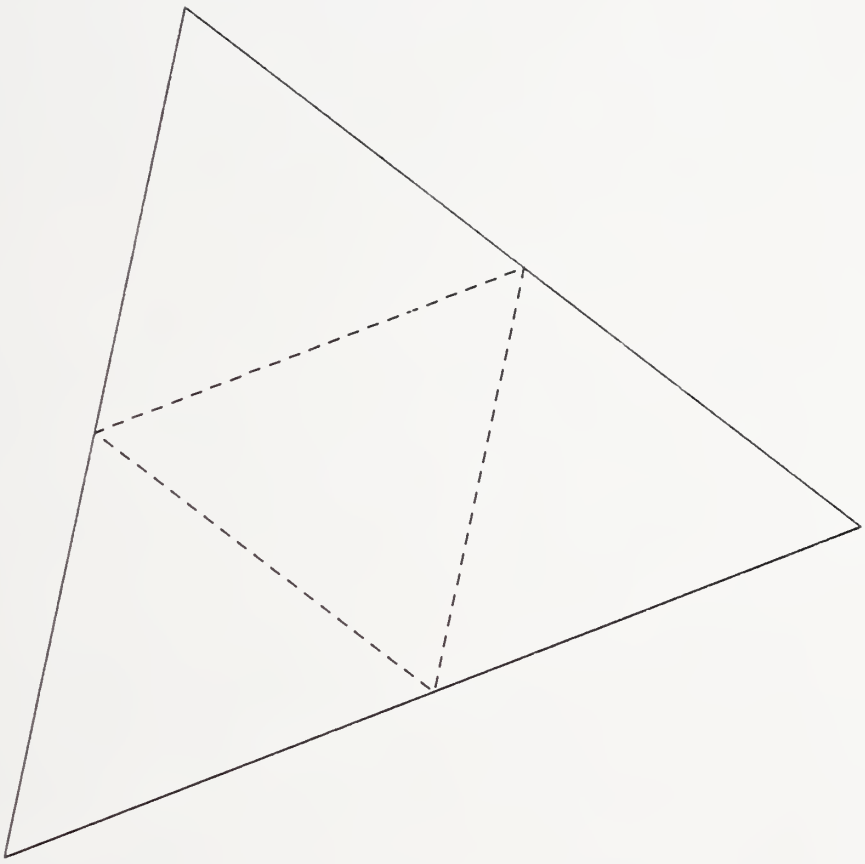

Figure 3. Division of a triangle into four similar triangles during mesh refinement.

For the refined triangulation, the previously calculated **u** is probably a good guess at the new **u**, so that few Newton iterations should be necessary. The hardest work is done on the initial triangulation, the one with the fewest vertices.

If there are two or more triangulations, the linear equations on the finest triangulation may be solved by a multilevel iteration [13, 14, 17], saving both time and storage space.

Using this procedure, the solution of eqs (1) and (2) is started by first obtaining a trial solution for the linearized form of eq (1), using a heuristic initial guess at the solution of eq (2) obtained from the one-dimensional charge-sheet model with the total charge as a boundary condition. The Jacobian of g required for the solution of eq (33) contains the Jacobian of eq (1). This Jacobian is approximated by

$$\frac{\partial g}{\partial u} = q\beta(n(y) + Q_{it} + Q_{otc}). \tag{37}$$

Once the surface charge and approximate local field term are obtained from the one-dimensional model and a local-field condition, the field term associated with the oxide is obtained using a fast Poisson solver [6]. The trial solution is then used to obtain a solution for eq (2) by quadrature and an additional fast Poisson solution in the oxide. This iterative procedure is repeated for each Newton step until the requested convergence is obtained or the specified maximum iteration count is reached.

4. Interactive Solution Strategies

The solution of the MOS transistor problem requires that a specific solution strategy be developed; finding an appropriate strategy is usually done with the interactive driver. Once the specific strategy has been determined, subsequent solutions can be obtained using a controlling subroutine with a preprogrammed strategy. This procedure has allowed us to solve, to any desired level of accuracy, all of the MOS transistor problems we have attempted. We have solved these problems starting from a rather simple initial guess, with no need for "sneaking up" on the solution through gradually changing device parameters or biases.

The problem presented here is a short-channel MOSFET. For all examples, we have used Shockley-Read-Hall recombination, field-dependent carrier mobilities, Fermi-Dirac statistics, $Q_{total} \approx 0$, and $\phi_{MS} = 0$.

The level 1 mesh used in these calculations is generated in two steps. First, a primitive mesh of the type shown in figure 4 is generated. The initial mesh is generated by specifying boundary segments which represent the geometry of the device. The present implementation of the program restricts device geometry to a planar device. This mesh restriction can be changed by rewriting the subroutine MESHIN. The minimum number of boundary segments is determined by the shape of the region of interest and by the boundary conditions. At each transition between boundary conditions types 1 and 2, a vertex is required.

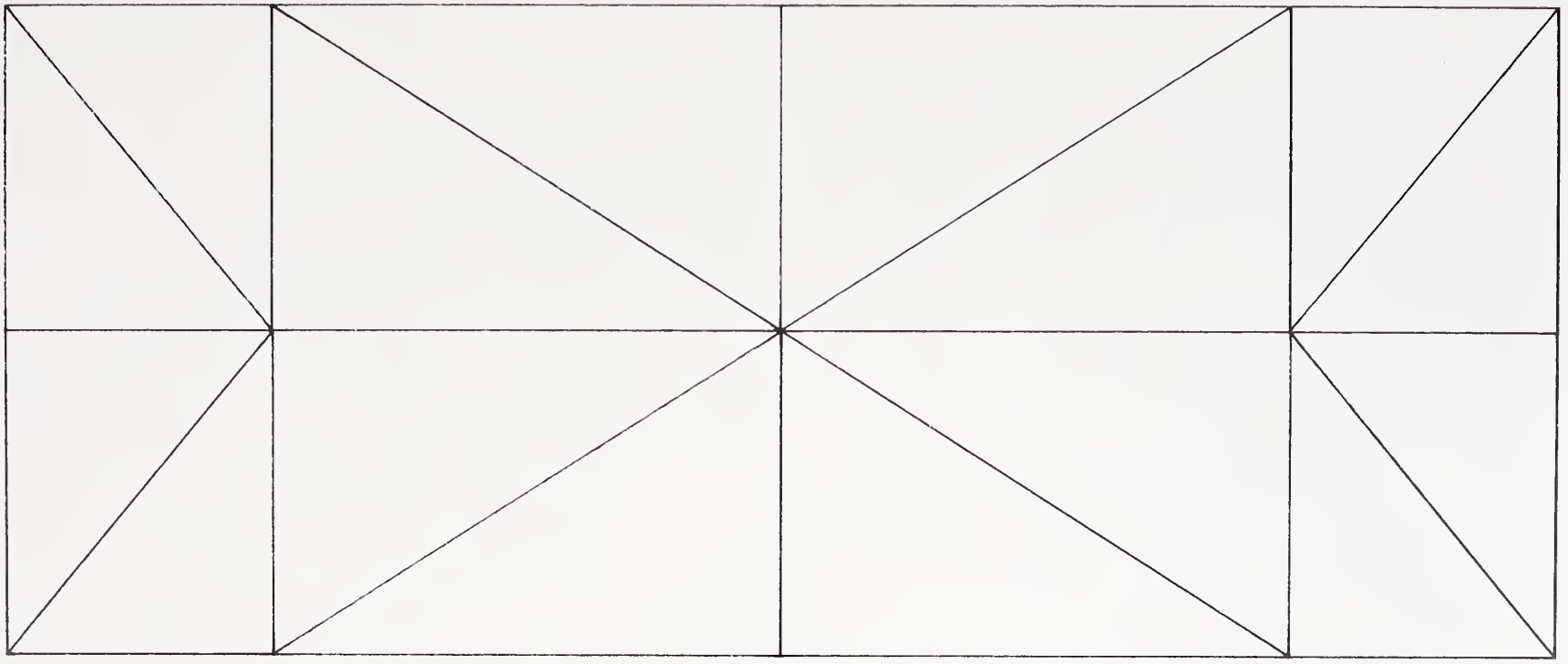

Figure 4. Primitive mesh for MOS transistor.

After the primitive mesh has been generated, user-specified prerefinements are used to place mesh points where the user thinks they are needed, based on physical understanding of the device. In the case shown in figures 4 and 5, the prerefinements were used to increase the mesh density in the inversion layer and in the depletion layer.

The difficulty of obtaining the level 1 solution is strongly dependent on the accuracy of the initial guess. The most primitive initial guess is to make the solution variables satisfy eqs (13), (14), and (15), with $\phi_{MS} = 0$ at each point in the device. This is the local space charge neutral condition. When Fermi statistics are used, eqs (16), (17), and (18) are used instead. The initial guess used in the MOS1 program is constructed by assuming values for the surface potential, based on a charge-sheet approximation, and assuming a shape for the bottom of the depletion edge in the transistor. These assumed values are then used to construct a two-dimensional approximation to the potential and quasi-Fermi levels based on simple p-n junction theory and the depletion approximation. In heavily doped material, local space charge neutrality is assumed; this applies in the source and drain. The two-dimensional fields around each junction are rotated one-dimensional abrupt p-n junctions. An idealized one-dimensional inversion layer is assumed in the channel. In any region below the channel where junction and inversion layer fields are both present, the larger of the two possible potentials is used. An initial guess of this type is shown in figure 6, and the final solution is shown in figure 7.

After an accurate solution of the nonlinear finite element equations for a particular level, which is characterized by small L_2 and L_∞ norms of the residuals, and the L_∞ norm of the Newton step less than kT/q (1 in unnormalized units), the discretization error in the solution can be reduced to any desired level of accuracy by refining the mesh adaptively, reducing the discretization error.

The strategy for obtaining the level 1 solution is usually significantly different from the strategy used at higher levels, because the initial guess usually has low accuracy. In devices where one of the carriers is only perturbing the potential, it is fastest to start the iteration process by solving the potential alone, then solving the potential and the dominant carrier, and finally solving all three carriers together.

The first Newton step taken on level 2 is critical. Since the solution is an exponential function of the change in potential and quasi-Fermi levels, the L_2 norm of the residual after taking this first step can be very large; numbers exceeding 10^{36} are not unusual. Small reductions in the size of the Newton step bring exponential reductions in the L_2 norms. At level 2 and higher levels, the error in the potential usually dominates the iteration. This seems to be caused by the more rapid variation of the right-hand side of this equation with solution variables.

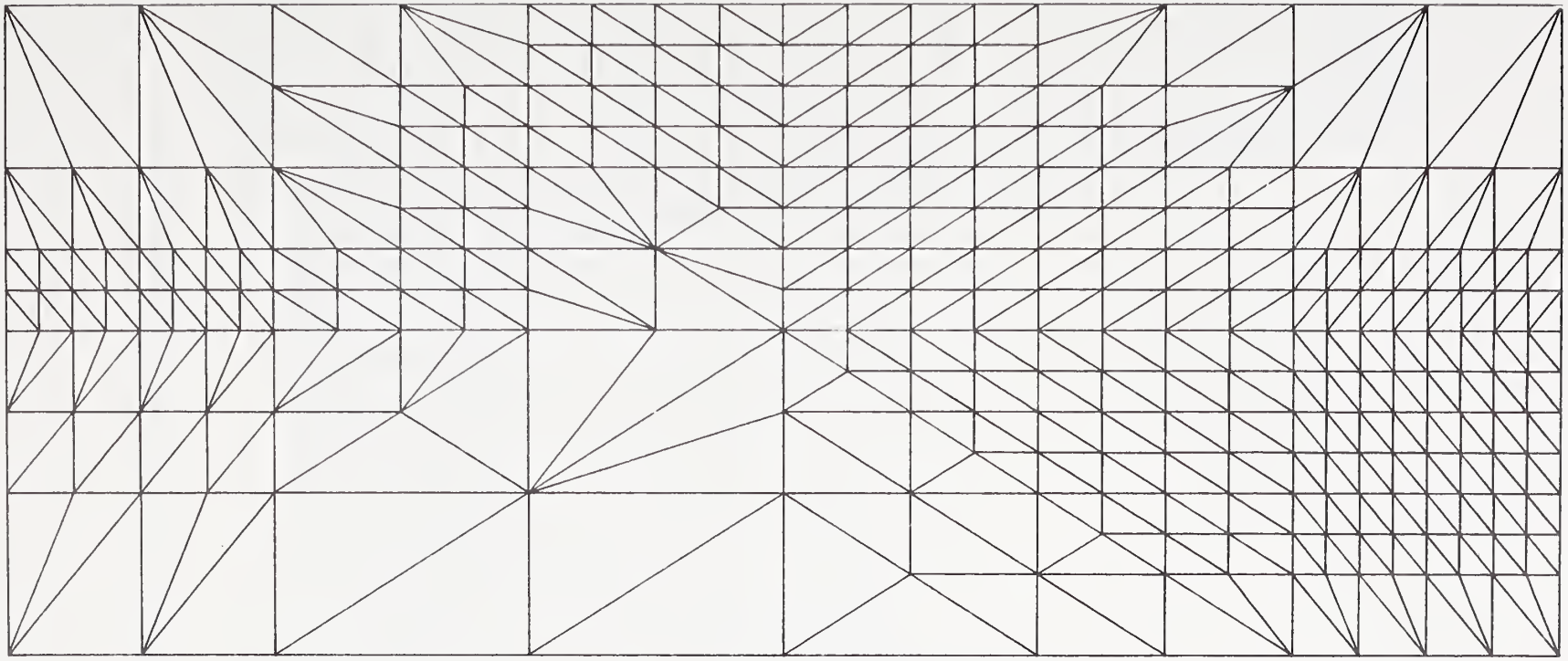

Figure 5. Level 1 mesh for MOS transistor.

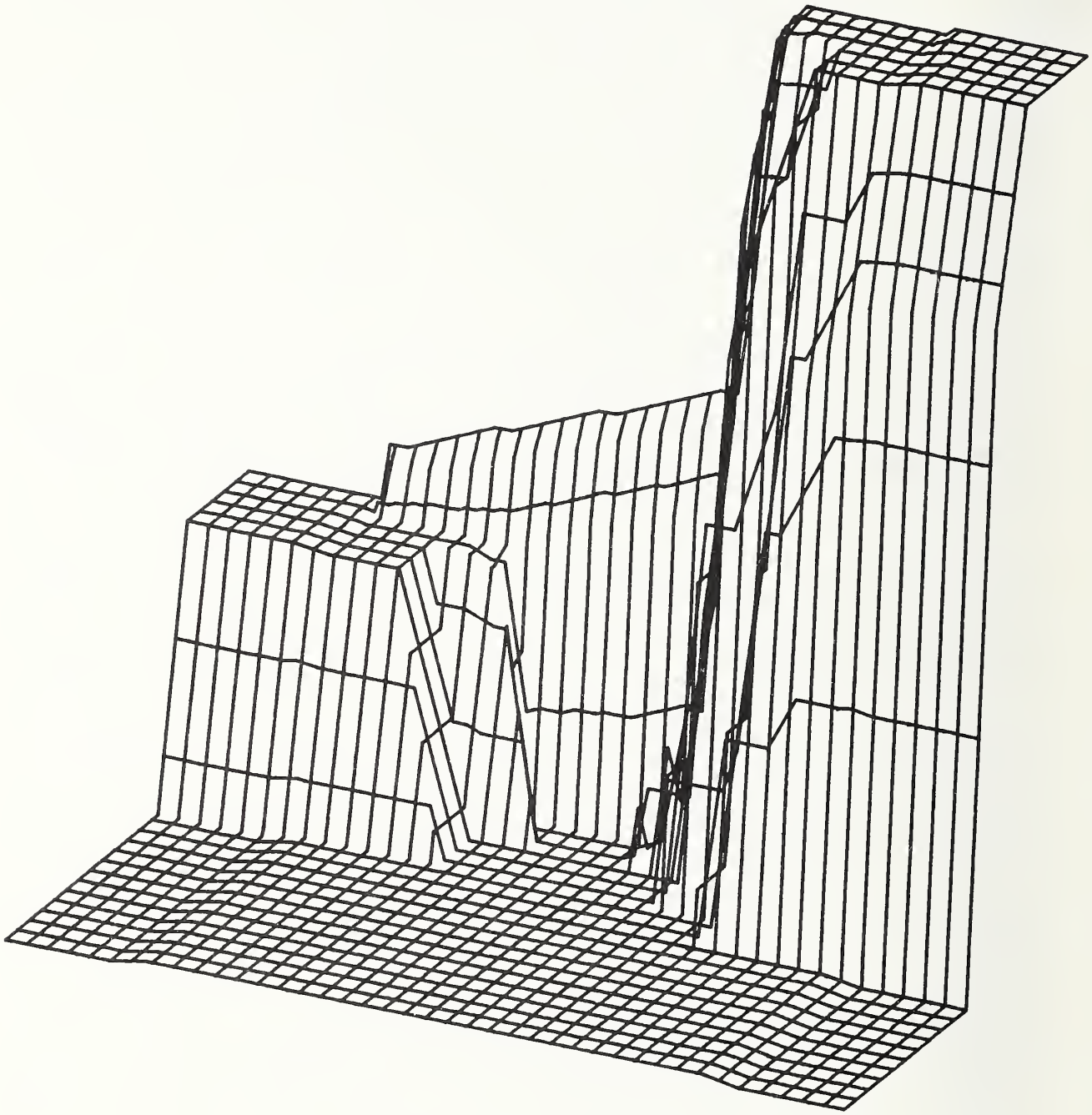

Figure 6. Initial guess for MOS transistor.

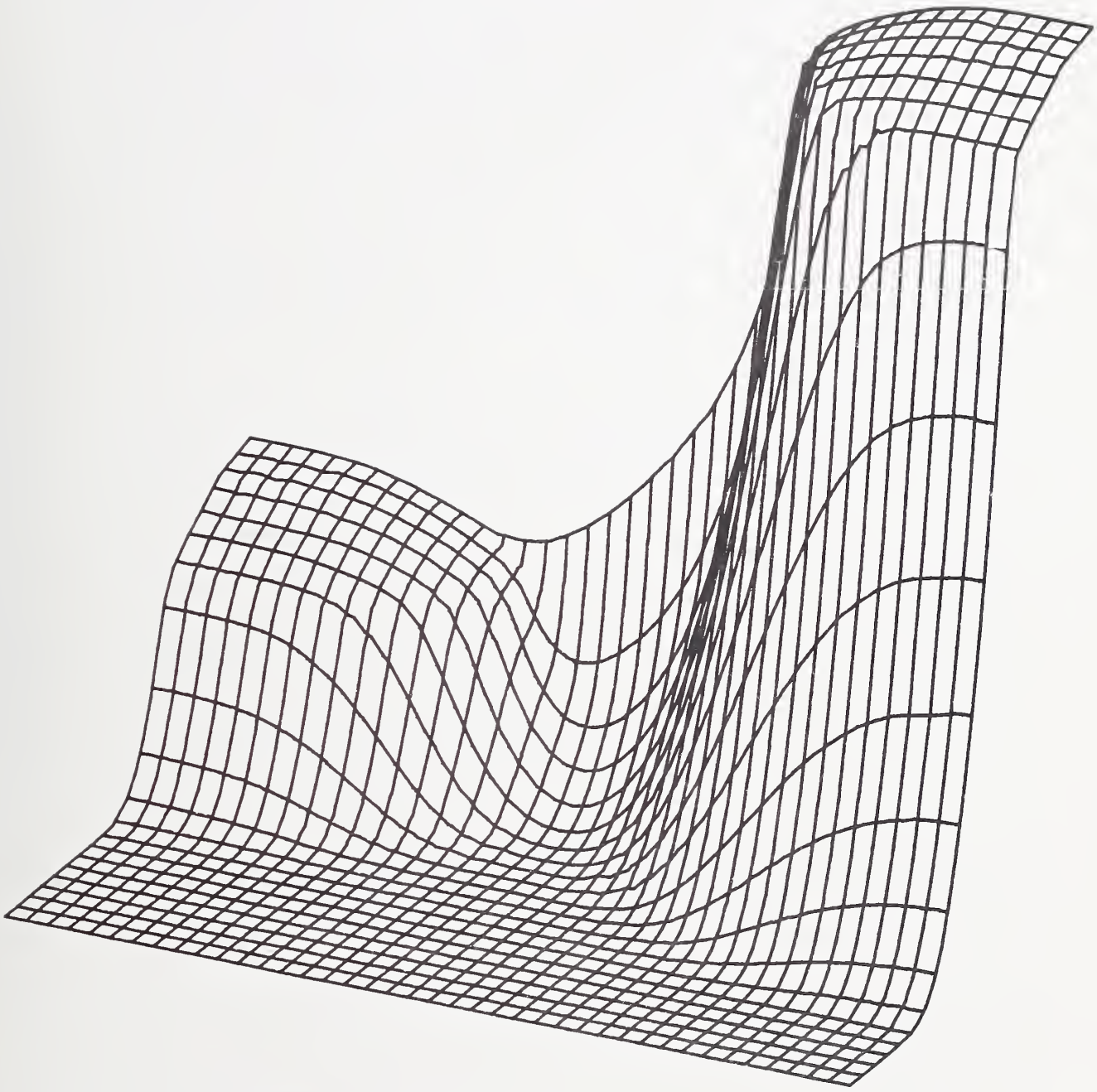

Figure 7. Final potential for MOS transistor.

Mesh refinements beyond level 2 are usually routine. All equations may be used and the pattern of refinement is such that solution norm problems are unusual. Convergence is rapid, usually five iterations or less. The quasi-Fermi levels converge more rapidly than the potential.

5. Program Specifics

The distributed source of the MOS1 program is divided into four files: MOS1.FOR, SYSTEM.FOR, B2DE.FOR, and DEVLIB.FOR. These four files conform to ANSI 77 FORTRAN IV and can be compiled on any FORTRAN compiler which meets this standard and accepts lower case input. The files MOS1.DOC, MOS1.EXP, and MOS1.LNK are also provided. MOS1.DOC provides a set of user instructions identical to Appendix A of this report. MOS1.EXP contains an example set of program input and output which is identical to Appendix B of this report. The file MOS1.LNK is a sample set of linkage control statements which have been used to link the program on the DEC VAX 11/780 under the VMS system version 3.7.

The file MOS1.FOR is the actual MOSFET model program. This file contains the subroutines which control input and output, model initial approximations, find the oxide field solution, and contain the calls to the subroutines which solve the partial differential equation. The oxide solution is calculated using the fast methods discussed in reference [6]; these subroutines are in the file DEVLIB.FOR. A complete copy of the Swarztrauber and Sweet package [6] is not included. Only those parts of the package that are used by MOS1 are included.

The file SYSTEM.FOR contains subroutines which are used in the other three program files but which are system dependent. The subroutines ISTIME and IFTIME are used to start and stop clocks which are used to measure the execution times of the parts of the partial differential solution process. If these routines are not used and dummy subroutines are substituted, ISTIME must be less than, NOT equal to, IFTIME. If these subroutines are dummy routines, then the timing information in the output summary will be incorrect.

The terminal-dependent programs in B2DE are set up to use Hewlett Packard HP26xx terminals. Users with other terminals will need to modify six programs: TRMCOL, TRMCRS, TRMDRW, TRMMOV, TRMSCL, and TRMPLT. The calling sequences and requirements are listed in the versions supplied.

The subroutines R1MACH and I1MACH are used to set machine constants which are used in the other program modules to set machine-dependent quantities. The original versions of the functions are discussed in reference [18] as are the procedures used to generate these constants for a type of computer which is not presently included in the present list. The conversion of the program from one computer type to another should not require modification of any part of the program which is not in the SYSTEM.FOR module. This assumes that some plotting package is available and that timing subroutines are available on the target computer system.

The module B2DE.FOR contains the subroutines used to solve the partial differential equations. The theory of these subroutines has been discussed in section 2.2 above and is presented in detail in [1].

The module DEVLIB.FOR contains subroutines used by the other three modules. These subroutines have been obtained from several sources. The error-handling software, memory management software, and machine-independent constant software were obtained from [18]. The fast elliptic solver was obtained from [6]. In the B2DE module, the Yale Sparse Matrix Package is used [16,19].

6. Examples

Two series of transistors are used to verify the accuracy of the model and parameter extraction methods discussed above. These transistors were selected to show the effect of variation of source-drain profile on transistors with submicron channel lengths and the effect of source-drain profile shape. In the first series, the source-drain profile is diffused phosphorus with a junction depth of 1.17 μm and channel lengths of 8.17, 1.83, and 0.80 μm. These channel lengths were obtained from SEM measurements of the polysilicon gate length and lateral diffusion as described in [20]. The second series of transistors has a source-drain profile of arsenic with a junction depth of 0.56 μm and channel lengths of 6.19, 4.29, 2.19, and 1.04 μm. Only the 2.19-μm device will be discussed.

6.1 Modeling of Phosphorus S-D Device

The phosphorus source-drain of the first series of transistors shows the effect of gradually doped source-drain transistors on the scaling of the field and currents. The simulation parameters of a typical device are given in Table 2.

Table 2.
Phosphorus S-D Device Simulation Parameters

L_{ch}	8.17	μm
Q_{it}	0.0	cm^{-2}
V_G, V_D	0.5V to 4.0V, 0.25V to 4.0V	
t_{ox}	0.0523	μm
N_{ch}	1.5×10^{15}	cm^{-3}
L_{jun}	0.93	μm
D_{jun}	1.17	μm

The measured subthreshold characteristic of the 8.17-μm transistor is shown in figure 8. The output characteristics of the same device are shown in figure 9. Good agreement between measured data and simulations has been obtained with no free parameters. The potential, electron density, and hole density in this device are shown as contour plots in figure 10 at a bias of $V_g = 2.0$ V and $V_d = 0.25$ V. Since this is a long-channel transistor, the electron density, shown in red, is parallel to the gate as expected. The hole density, shown in green, decreases entering the depletion region from below. The potential, electron density, and hole density are shown as contour plots in figure 11 at a bias of $V_g = 2.0$ V and $V_d = 2.0$ V. This figure shows the expansion of the depletion region into the body caused by increasing the drain voltage.

The measured subthreshold characteristic of the 1.83-μm transistor is shown in figure 12. The output characteristics of the same device are shown in figure 13. Good agreement between measured data and simulations are still obtained with no free parameters. The potential, electron density, and hole density in this device are shown as contour plots in figure 14 at a bias of $V_g = 2.0$ V and $V_d = 0.25$ V. The hole density, shown in green, decreases entering the depletion region from below. The potential contours are parallel to the gate in the channel in this device, but most of the depletion region charge is controlled by the source-and-drain fields since the bottom edge of the depletion region is curved along its entire length. The potential, electron density, and hole density are shown as contour plots in figure 15. This figure shows the expansion of the drain-field-controlled part of the channel caused by increasing the drain voltage. This results in lowering the output impedance of the transistor as seen by comparing figures 13 and 9.

The output characteristics of the 0.80-μm transistor are shown in figure 16. Even in the triode region of a submicron transistor, good agreement between measured data and simulations has been obtained with no free parameters. The potential, electron density, and hole density in this device are shown as contour plots in figure 17 at a bias of $V_g = 2.0$ V and $V_d = 0.25$ V. At this bias, the potential contours are curved at each point in the channel, indicating that every point in the channel is affected by the drain field. The potential, electron density, and hole density are shown as contour plots in figure 18 at a bias of $V_g = 2.0$ V and $V_d = 2.0$ V. This figure shows the expansion of the depletion region into the body caused by increasing the drain voltage, causing the electron flow in the channel to expand into the bulk. The potential, electron density, and hole density are shown as contour plots in figure 19 at a bias of $V_g = 2.0$ V and $V_d = 3.0$ V. At this voltage, significant numbers of electrons are found 0.5 μm below the oxide-silicon interface. The density of these carriers is high enough to modulate the channel charge but not high enough to carry significant current.

This sequence of model calculations allows the physical mechanisms involved in the short-channel effect to be clearly separated. At all channel lengths below 8.0 μm, the drain field is affecting the channel and modulating the output impedance of the transistors. These changes in output impedance are NOT caused by mobility effects since the fields in the channel are below the critical values shown in figure 2, but are the result of drain field channel length modulation. When the drain field has penetrated the entire channel, the usual short-channel subthreshold characteristics are observed. These are adequately modeled by the two-dimensional model as is demonstrated by the data in figure 16.

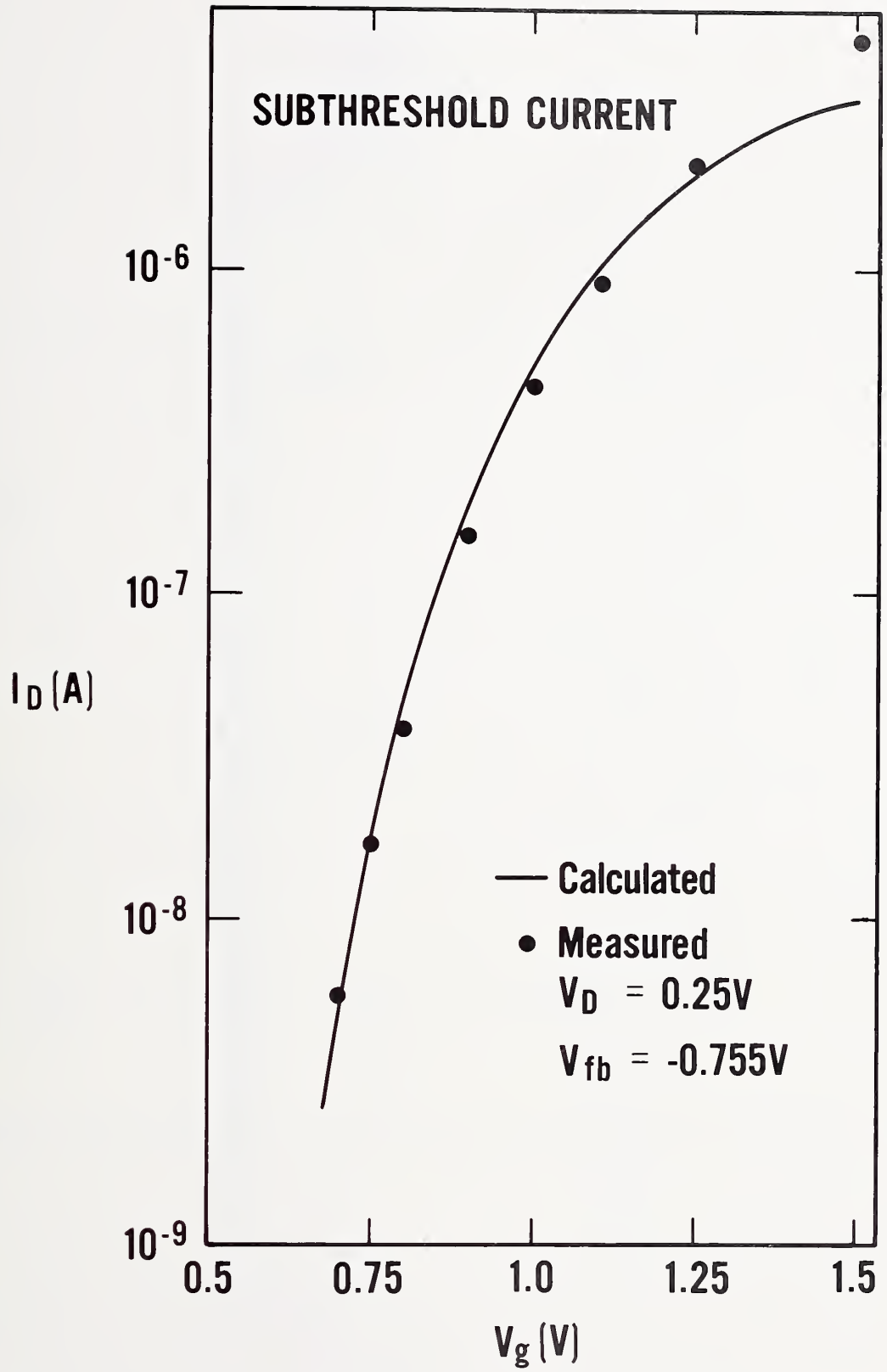

Figure 8. The measured subthreshold characteristic of the 8.17-μm transistor.

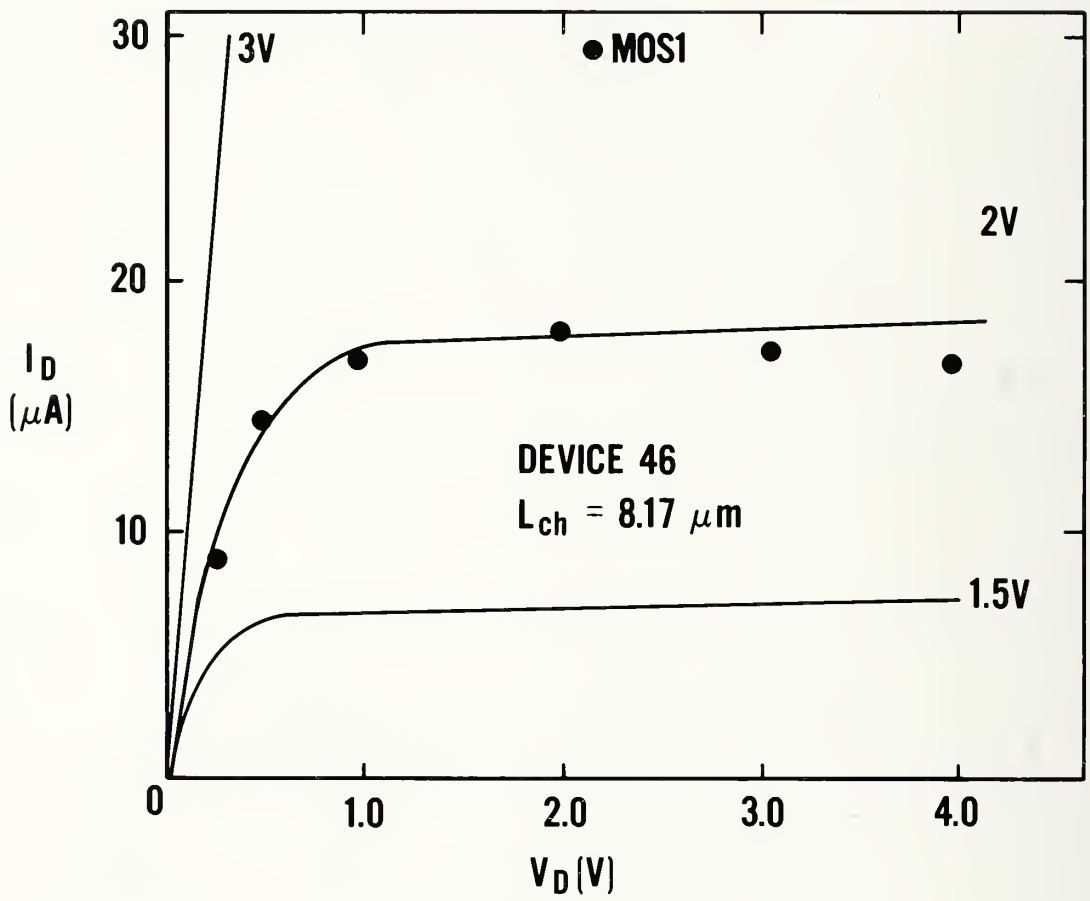

Figure 9. The measured output characteristics of the 8.17 – μm transistor.

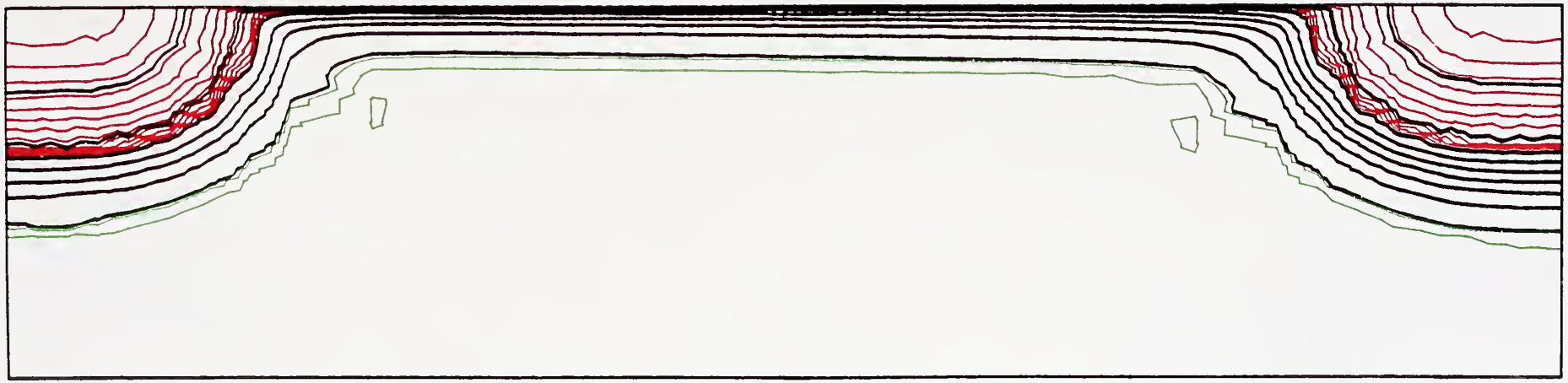

Figure 10. The potential, electron density, and hole density in the 8.17 − μm transistor at a bias of $V_g = 2.0$ V and $V_d = 0.25$ V.

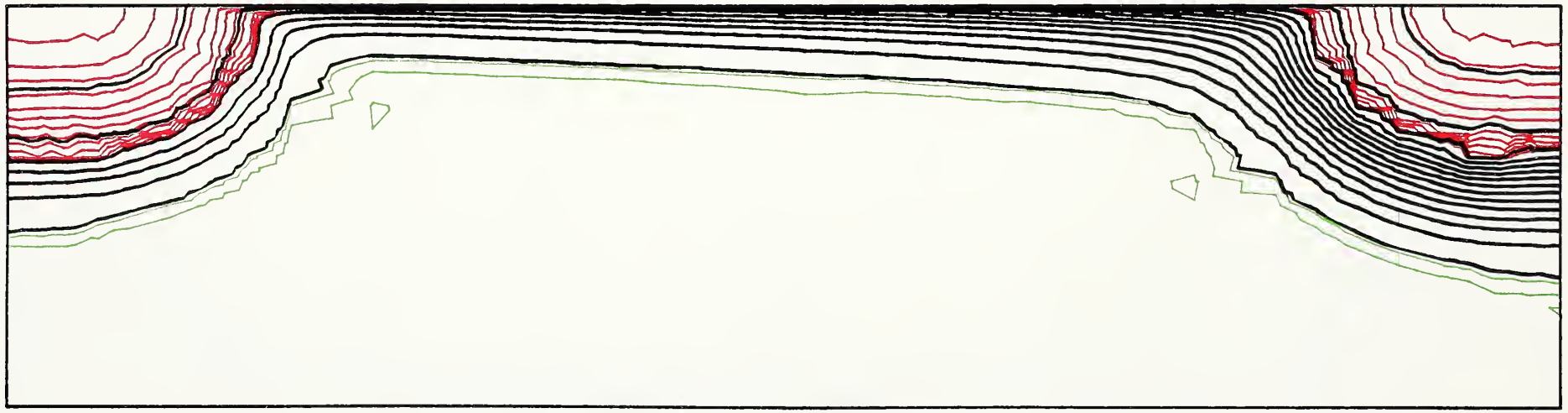

Figure 11. The potential, electron density, and hole density in the $8.17 - \mu$m transistor at a bias of $V_g = 2.0$ V and $V_d = 2.0$ V.

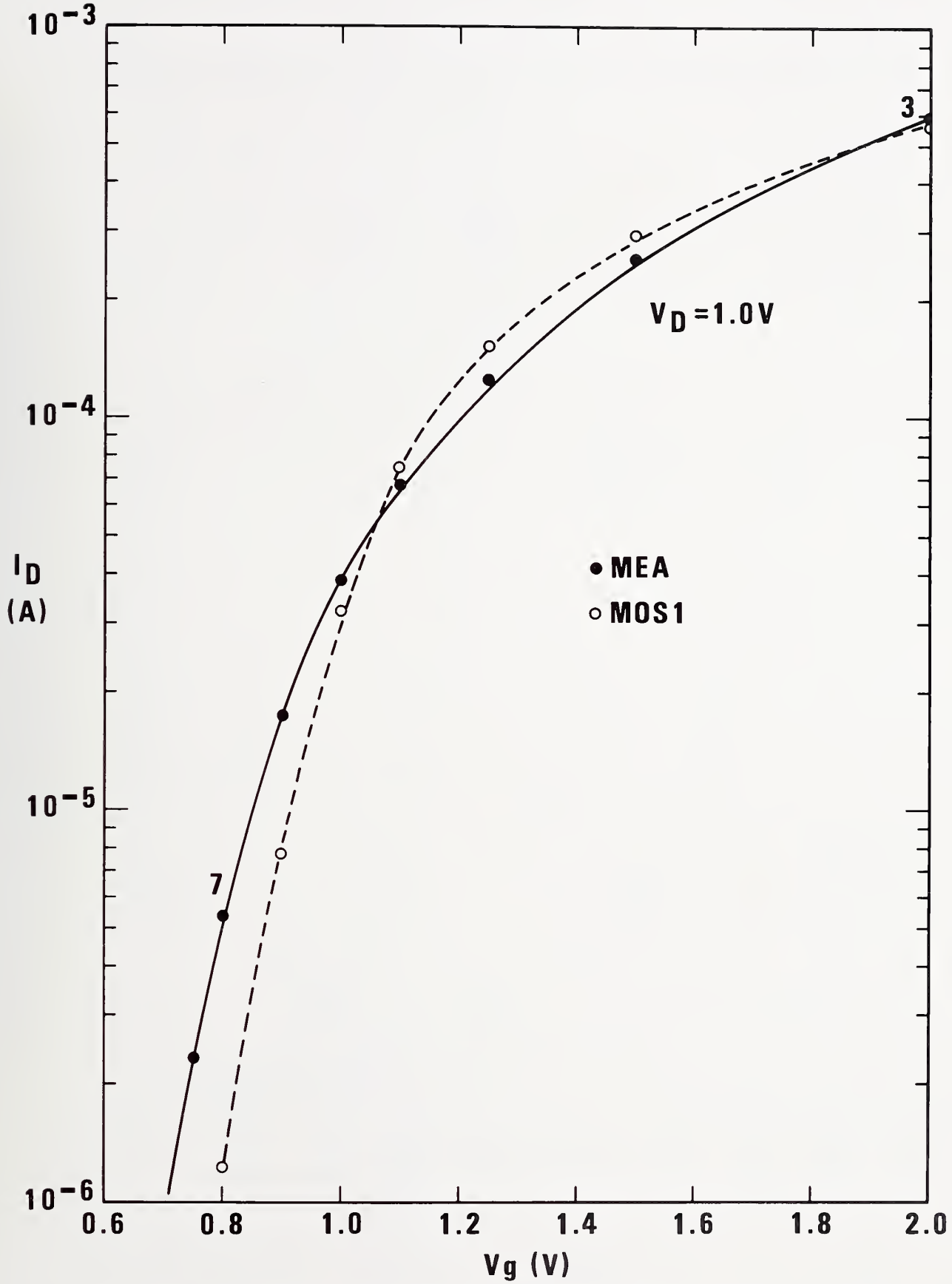

Figure 12. The measured subthreshold characteristic of the 1.83-μm transistor.

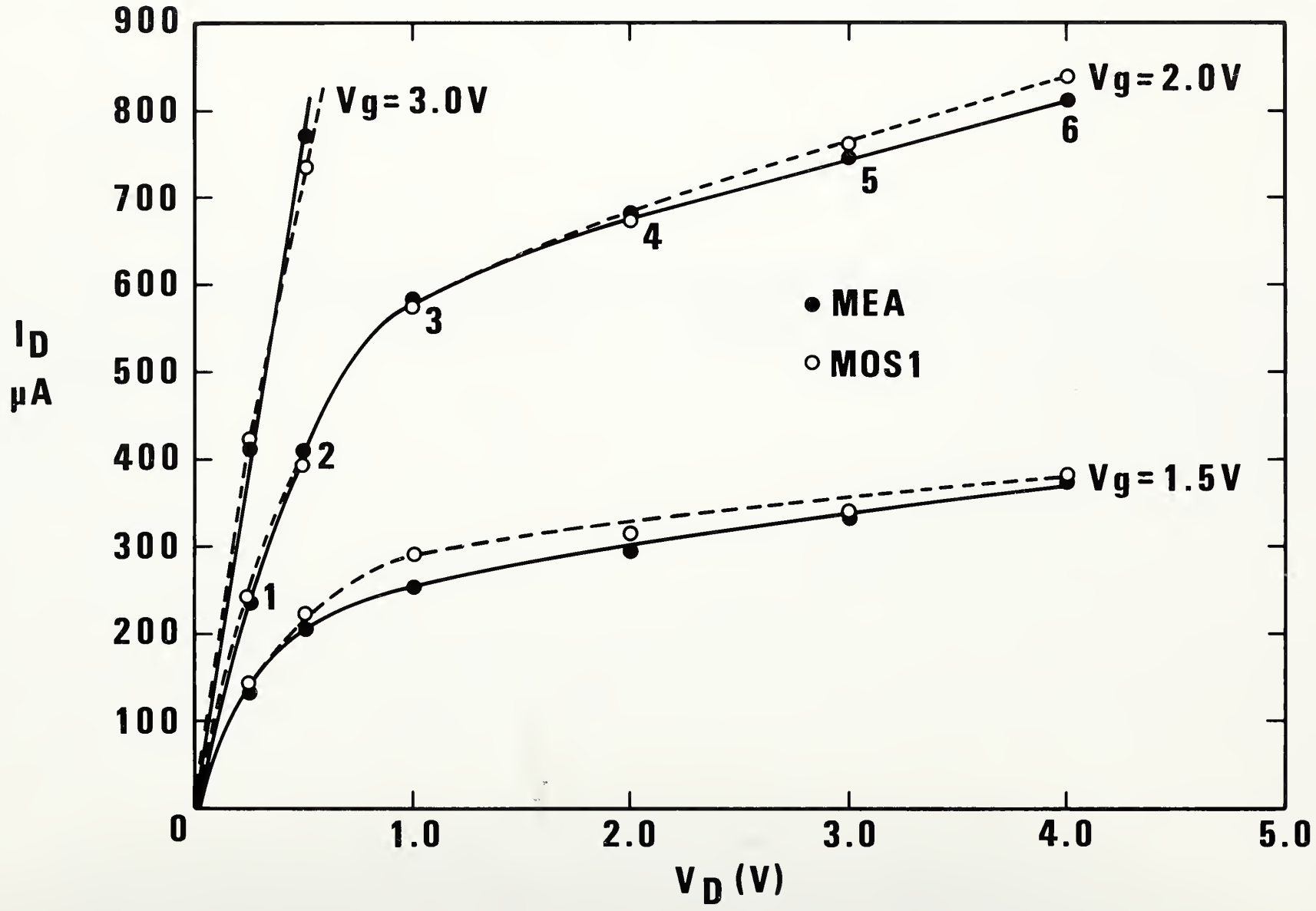

Figure 13. The measured output characteristics of the 1.83 – μm transistor.

Figure 14. The potential, electron density, and hole density in the 1.83 − μm transistor at a bias of $V_g = 2.0$ V and $V_d = 0.25$ V.

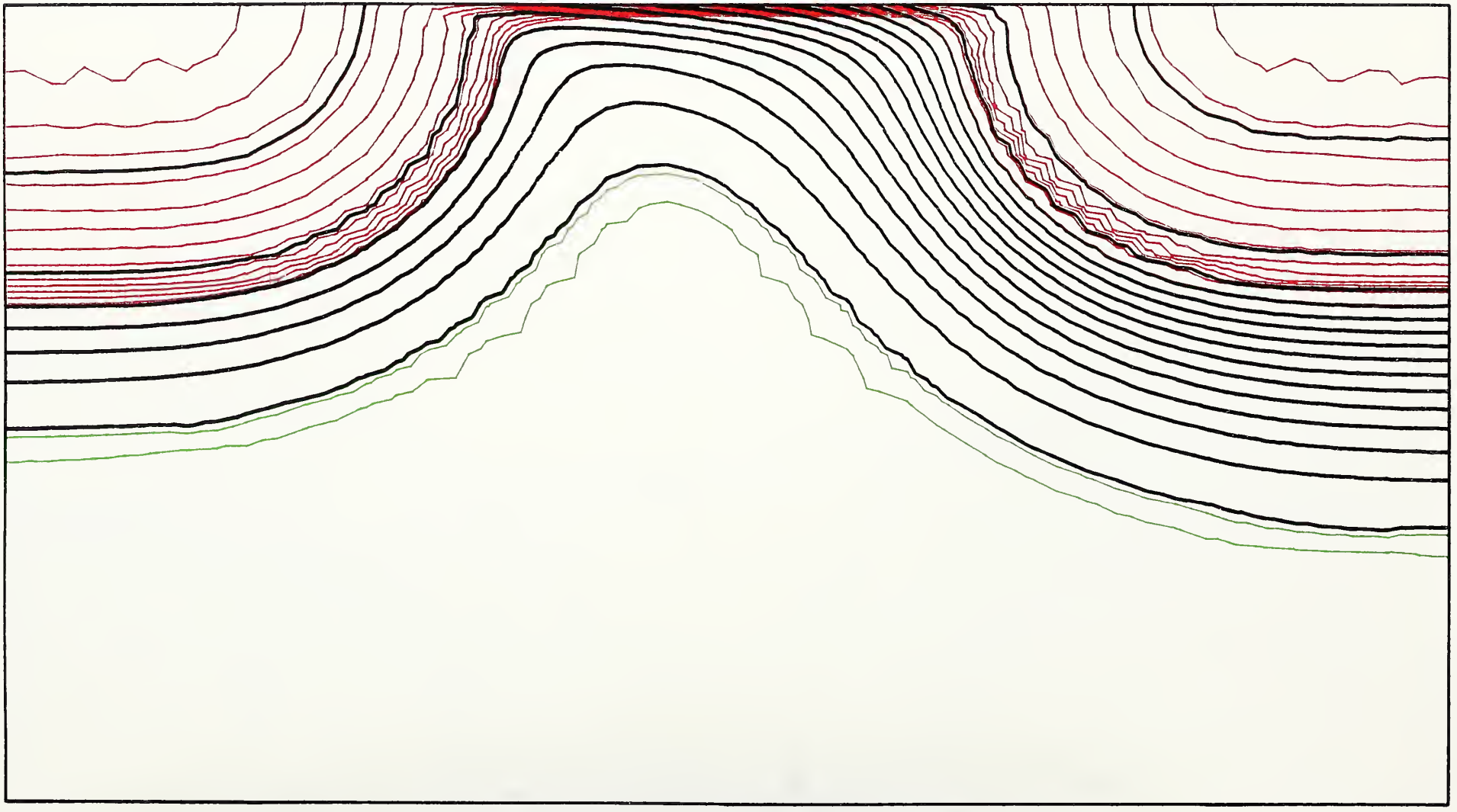

Figure 15. The potential, electron density, and hole density in the $1.83 - \mu$m transistor at a bias of $V_g = 2.0$ V and $V_d = 2.0$ V.

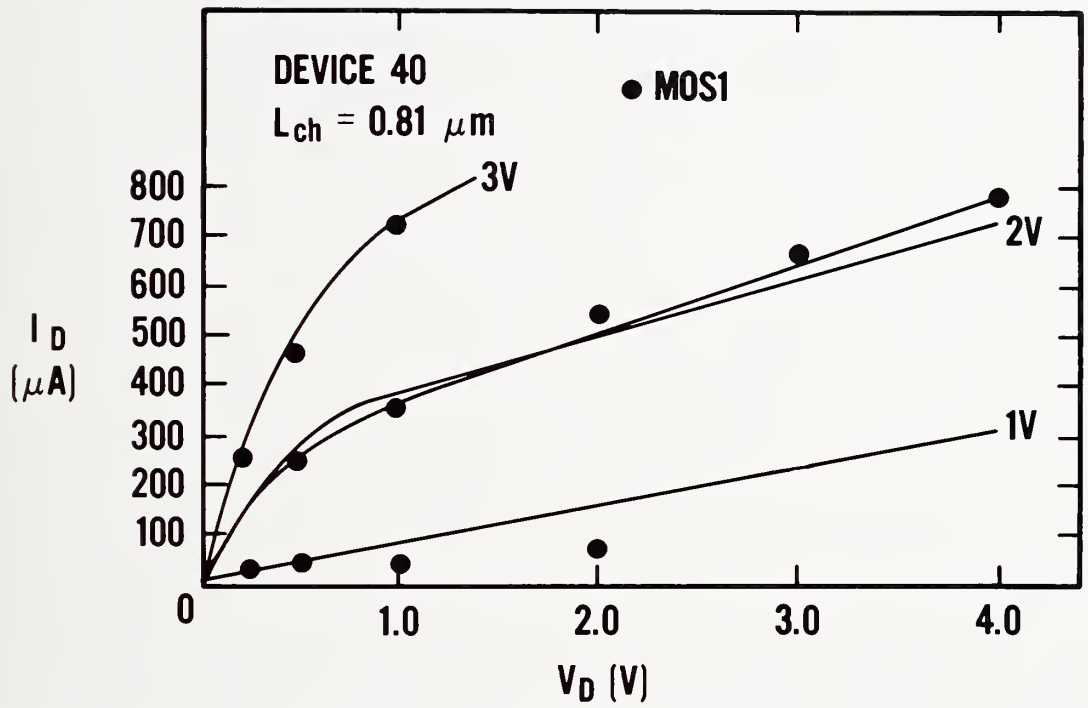

Figure 16. The measured output characteristics of the 0.80 – μm transistor.

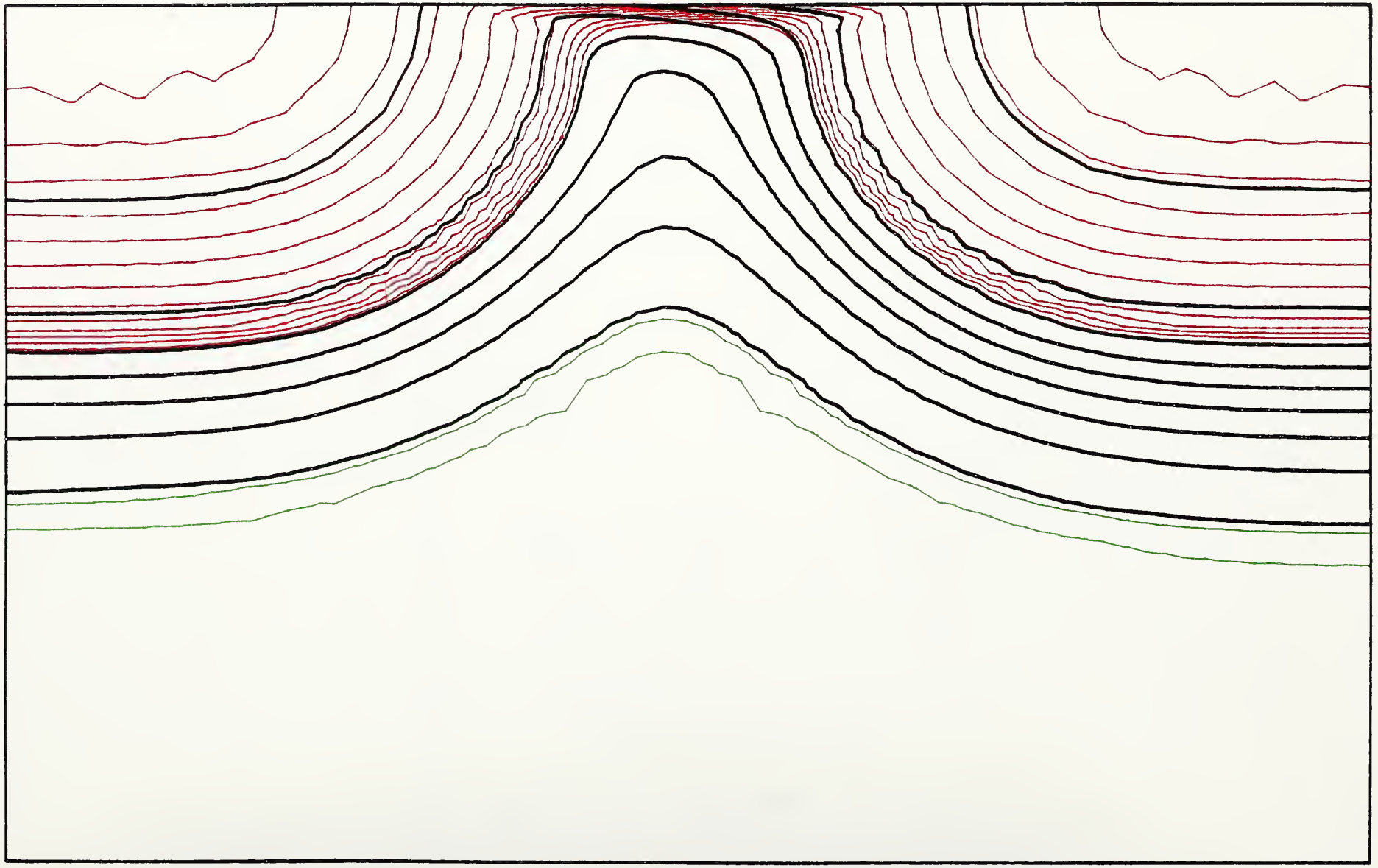

Figure 17. The potential, electron density, and hole density in the $0.80 - \mu$m transistor at a bias of $V_g = 2.0$ V and $V_d = 0.25$ V.

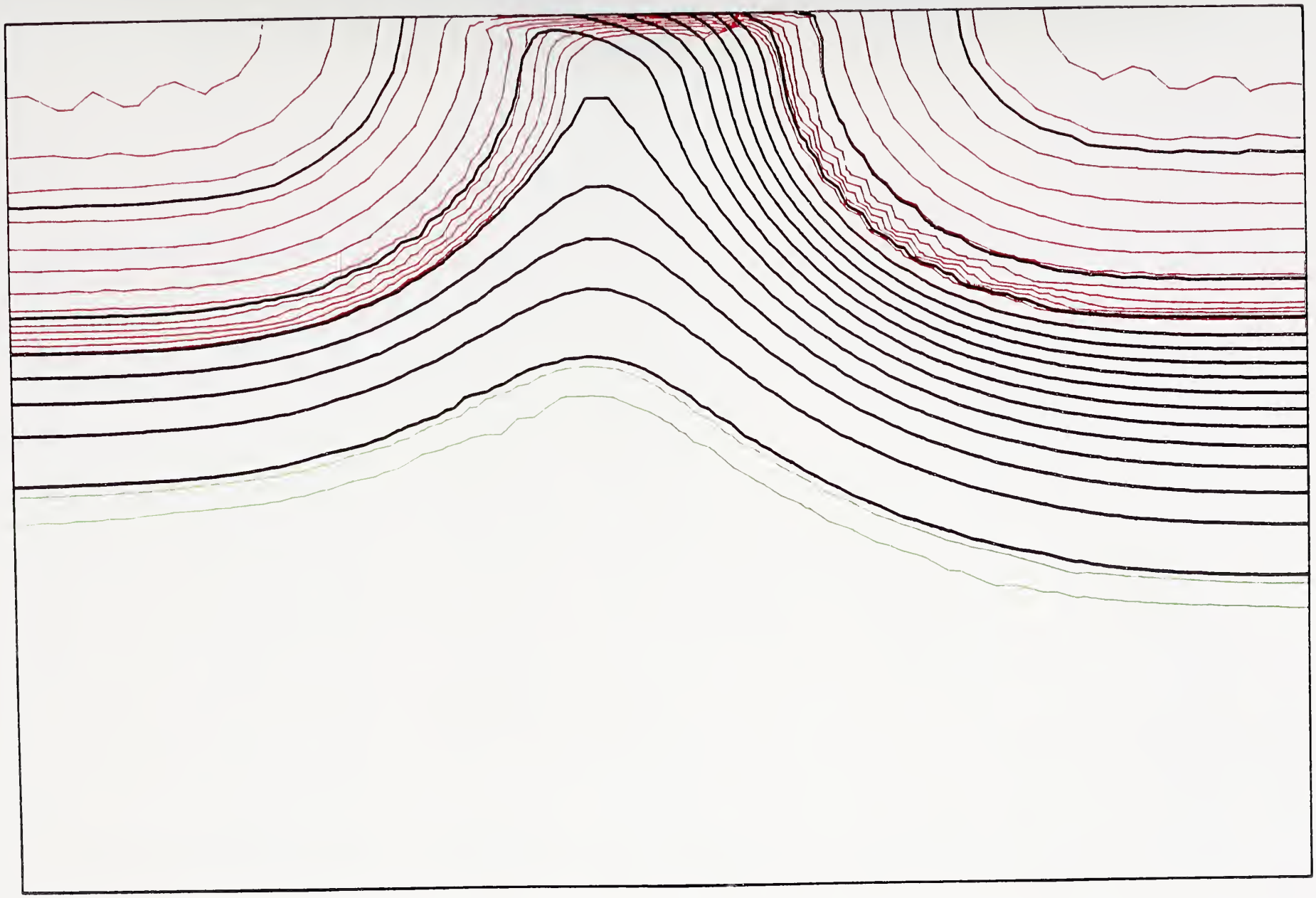

Figure 18. The potential, electron density, and hole density in the $0.80 - \mu$m transistor at a bias of $V_g = 2.0$ V and $V_d = 2.0$ V.

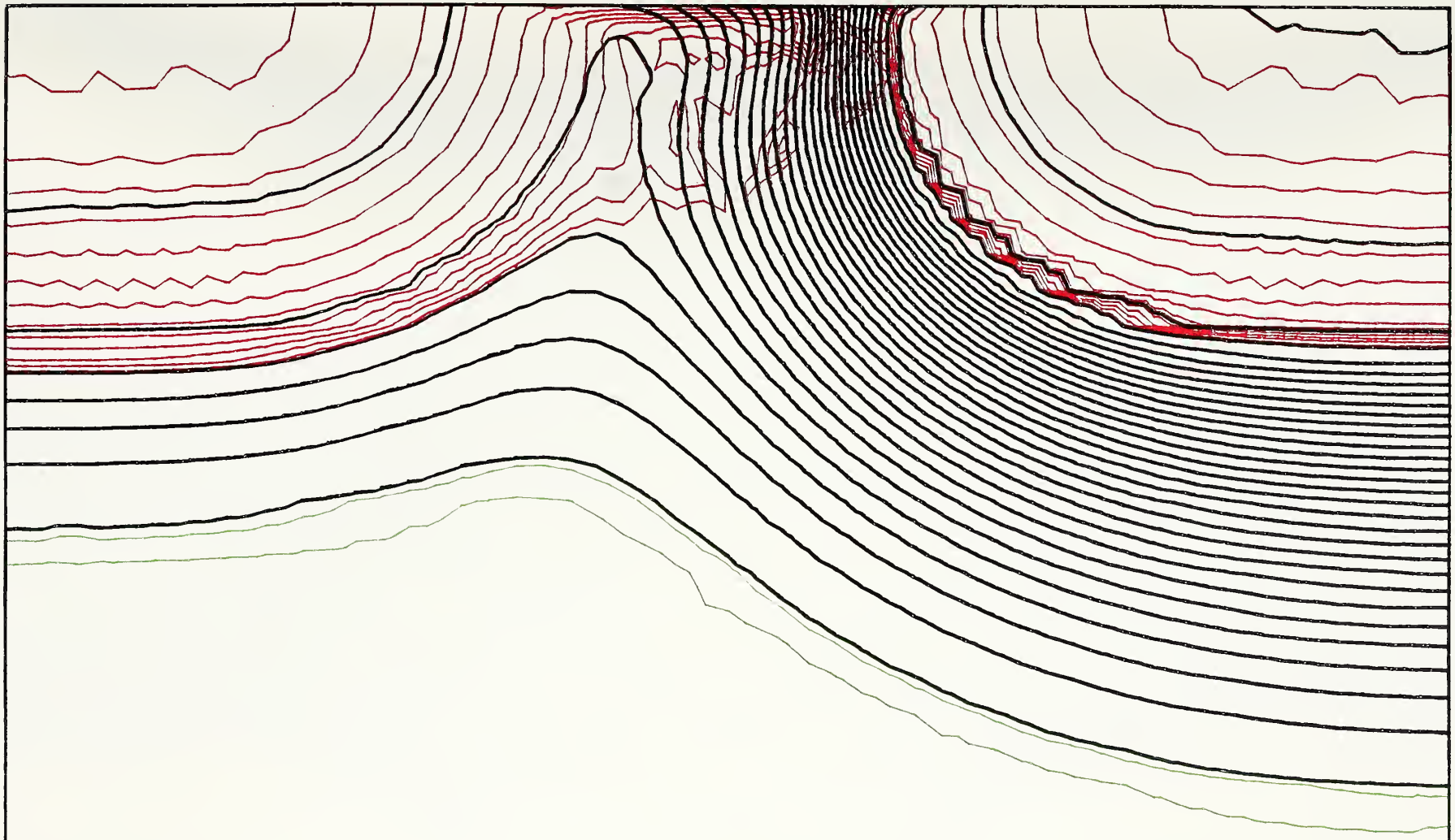

Figure 19. The potential, electron density, and hole density in the 0.80 – μm transistor at a bias of $V_g = 2.0$ V and $V_d = 3.0$ V.

Finally, after the saturation and subthreshold regions are dominated by short-channel effects, the triode region current is altered by the dual effects of velocity saturation and field distortion in the channel caused by spreading of the electrons into the depletion region. These two effects are strongly coupled and have not been seen except as a combined effect.

6.2 Modeling of Arsenic Device

The simulation parameters for a typical arsenic device are given in table 3.

Table 3.
Arsenic S-D Device Simulation Parameters

L_{ch}	2.19	μm
Q_{it}	0.0	cm^{-2}
V_G, V_D	0.5V to 4.0V, 0.25V to 4.0V	
t_{ox}	0.0635	μm
N_{ch}	1.38×10^{15}	cm^{-3}
L_{jun}	0.38	μm
D_{jun}	0.56	μm

The measured subthreshold characteristic of the 2.19-μm transistor is shown in figure 20. The output characteristics of the same device are shown in figure 21. Good agreement between measured data and simulations has been obtained with no free parameters. The source-drain doping profile was modeled using an abrupt impurity gradient model approximating the profile from [8] with junction depths calibrated to the measurements in [12].

In devices with channel lengths at or below 1.0 μm, the drain current is dominated by drain field modulation and velocity saturation. At these channel lengths, the details of the source-drain profile are not important. In the intermediate region when the drain field penetration of the channel is incomplete, the shapes of the source-and-drain profiles are critical.

7. Summary

We have developed a capability for analyzing MOS transistors in two dimensions, using a general-purpose PDE box, B2DE. We have been able to solve a wide variety of MOSFETs under several different sets of operating conditions. We have shown that these methods work for all regions of device operation, nonideal oxide interfaces, and for either Boltzmann or Fermi-Dirac statistics. When accurate values of the input model parameters are available, the model accurately reproduces measured current-voltage characteristics of sub-micron transistors.

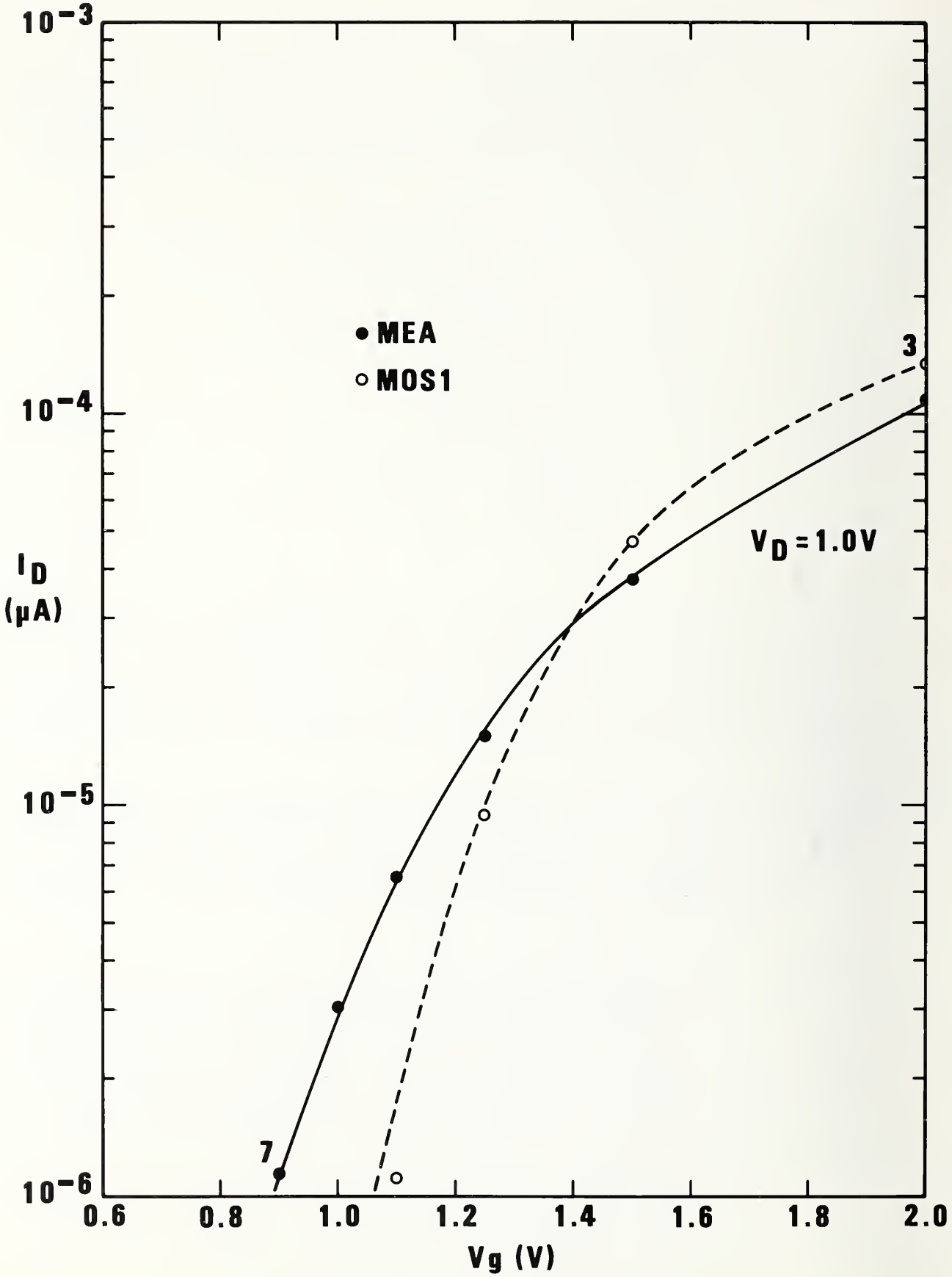

Figure 20. The measured subthreshold characteristic of the 2.12-μm transistor.

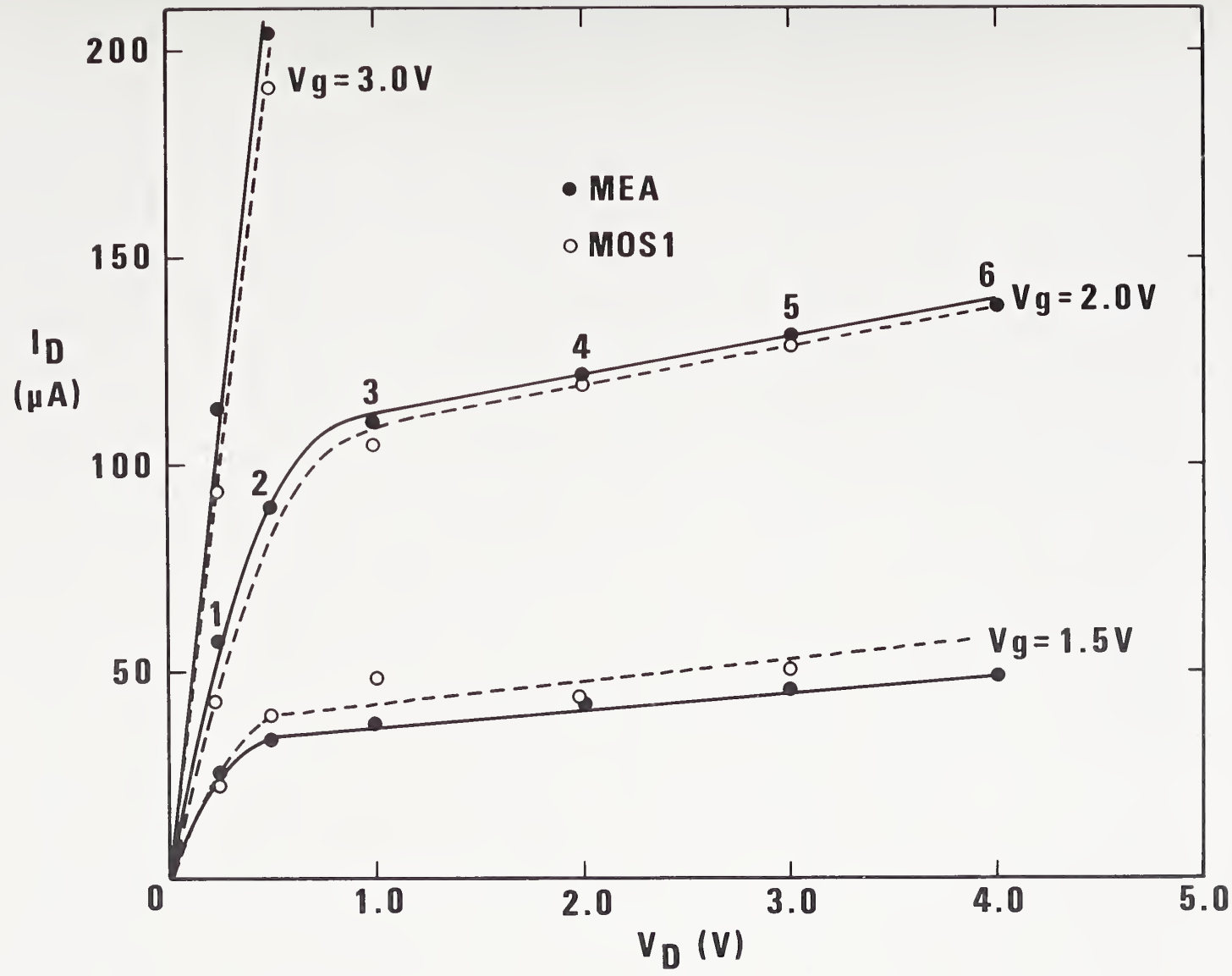

Figure 21. The measured output characteristics of the 2.12 – μm transistor.

References

[1] J. L. Blue, "B2DE-A Program for Solving Systems of Nonlinear Elliptic Partial Differential Equations in Two Dimensions," in preparation.

[2] S. M. Sze, *Physics of Semiconductor Devices* (John Wiley and Sons, New York, 1981), pp. 16-57.

[3] K. K. Thornber, "Current Equations for Velocity Overshoot," *IEEE Electron Device Letters* EDL-3, 69-71 (1982).

[4] J. A. Cooper and D. F. Nelson, "Measurement of High-Field Drift Velocity of Electrons in Inversion Layers of Silicon," *IEEE Electron Device Lett.* EDL-2, 171-173 (1981).

[5] J. R. Brews, "A Charge-Sheet Model of the MOSFET," *Solid-State Electronics* 21, 345-355 (1979).

[6] P. Swarztrauber and R. Sweet, "Efficient FORTRAN Subprograms for the Solution of Elliptic Partial Differential Equations," NCAR Technical Note, NCAR-TN/IA-109 (1975).

[7] D. P. Kennedy and R. R. O'Brien, "Analysis of the Impurity Atom Distribution Near the Diffusion Mask for a Planar p-n Junction," *IBM J. Res. Dev.* 9, 179-186 (1965).

[8] D. D. Warner and C. L. Wilson, "Two-Dimensional Concentration Dependent Diffusion," *Bell Sys. Tech. J.* 59, 1-41 (1980).

[9] W. Fichtner, E. N. Fuls, R. L. Johnston, T. T. Sheng, and R. K. Watts, "Experimental and Theoretical Characterization of Submicron MOSFETs," *Tech. Digest, 1980 International Electron Devices Meeting*, Washington, D.C., December 8-10, 1980, pp. 24-27.

[10] R. Fair, "Concentration Profiles of Diffused Dopants in Silicon," *Applied Solid State Science*, Suppl. 2B, Dawon Kahng, Ed. (Academic Press, 1981), pp. 1-108.

[11] P. Roitman, J. Albers, and D. R. Myers, "An Investigation of the Two-Dimensional Shape of Ion-Implanted Regions," *J. Appl. Phys.* 55, 4436-4443 (1984).

[12] J. Albers, P. Roitman, and C. L. Wilson, "Verification of Models for Fabrication of Arsenic Source-Drains in VLSI MOSFETs," *IEEE Trans. Electron Devices* 30, 1453-1462 (1983).

[13] R. E. Bank and A. H. Sherman, "PLTMG Users' Guide," CNA 152, Center for Nu-

merical Analysis, The University of Texas at Austin (September 1979).

[14] R. E. Bank, "PLTMG Users' Guide, June, 1981 version," Technical Report, Department of Mathematics, University of California at San Diego (August 1982).

[15] G. Strang and G. J. Fix, *An Analysis of the Finite Element Method* (Prentice-Hall, Englewood Cliffs, New Jersey, 1973).

[16] A. H. Sherman, "Algorithms for Sparse Gaussian Elimination with Partial Pivoting," *ACM Trans. Mathematical Software* 4, 330-338 (1978).

[17] A. Brandt, "Multi-Level Adaptive Solutions to Boundary-Value Problems," *Math. Comput.* 31, 333-390 (1977).

[18] P. A. Fox, A. D. Hall, and N. L. Schryer, "The PORT Mathematical Subroutine Library,"*ACM Trans. Mathematical Software* 2, 104-126 (1978).

[19] S. C. Eisenstat, M. C. Gursky, M. A. Schultz, and A. H. Sherman, Yale Sparse Matrix Package I. The Symmetric Codes, Research Report #112, Department of Computer Science, Yale University (1976).

[20] C. L. Wilson, P. Roitman, and J. L. Blue, "High Accuracy Physical Modeling of Submicron MOSFETs, " to be published in *IEEE Trans. Electron Devices.*

Appendix A
File MOS1.DOC

1. Interactive Execution

The program is executed by entering:

RUN MOS1.EXE

A prompt will appear for the next input;

Enter 0-new solution, 1-old solution, 2-quit

0-new solution

Set up inputs for a new problem.

1-old solution

Reads an old solution from the file insol.dat.

2-quit

Exits from program.

Enter 0-interactive, 1-batch

0-interactive

Set control variables for interactive running.

1-batch

Set control variables for batch running.

The main control menu then is printed.

1-print, 2-plot, 3-solve, 4-comp, 5-save, 6-end

1-print

prints outputs. Discussed below in detail in 2.

2-plot

plots outputs. Discussed below in detail in 2.
If the value of this parameter is an integer of
the form 12,22,...,62 then the first equation
is transformed into one of:

12-natural log of the electron density.

22-natural log of the hole density.

32-arc sinh of the net ionized impurity density.

42-arc sinh of the recombination-generation.

52-arc sinh of the net charge.

62-energy density transferred to electrons.

3-solve

Requests number of solution grid levels.
Number of levels. (Must be 8 or less)

4-comp

Allows solutions to be compared within a run.

5-save

Writes a solution on the file outsol.dat

6-end

exits to the first menu discussed after print current histogram.
This then causes a return too:

Enter 0-new solution, 1-old solution, 2-quit

All of the following files will be used by the program.

input.dat
plot.dat
doping.dat
insol.dat
outsol.dat

scrtch.dat

The first and third files must have been created prior to this execution. They contain input data used by the program. The second file may be created during execution.

A. FILES

1) input.dat contains data for each point to be calculated. The file format is shown below.
2) plot.dat is an empty file for output of plotting data if the plots are to be saved.
3) doping.dat contains a table of channel doping profiles.
4) insol.dat contains a binary image of a previous solution.
5) outsol.dat an empty file for a binary image of a solution.
6) scrtch.dat an empty file for intermediate storage.

B. Examples of FILE FORMATS.

Each point to be calculated requires 5 lines of input. Data on a given line are free-format. Comments may appear after all required data fields. After each point is calculated, the program will look for additional data until a source doping of zero is found (first line of input), after which the program will terminate.

EXAMPLE OF input.dat: Data are in units given.

```
Line 1:
Source doping/cm**3, Source and Drain profile type (1-3)
3.0e19 1
Profile types are 1-diffused, 2-abrupt diffused, and planted Gaussian.
Line 2:
Vertical junction depth(um),Lateral junction depth(um),Poly-Si length(um)
note:  Channel length = Poly-Si length-2*Lateral junction depths
0.55 0.84 5.00
Line 3:
Drain(volts),Gate(volts),Substrate(volts)
note:  Source voltage = 0.0.
0.15  +1.158 0.0
Line 4:
Oxide thickness(um),Oxide dielectric constant,Interface charge(cm**2)
0.057 4.0 0.0
Line 5:
```

```
Channel mobility(cm**2/V-sec),Channel width(um),Flatband voltage(V)
400.0 20.0 -.580
Line 6:
Line 6 may contain 0.0 for stopping program, or may be
first line of data for next point to be calculated.  Last line in
data file must be zero.

EXAMPLE of doping.dat:  Data for doping density.
Line 1:
Contains integer, Ndop, denoting the length of the profile table.
Lines 2 through Ndop+1:
Each contains an x (um) and the doping density at x (1/cm**3)
0.0 3.e15
100.0 3.e15
```

C. OUTPUT DISPLAY

All outputs (except the drain current) are in intrinsic Debye lengths (33um) and thermal volts (25mV), or units derived from these, such as Thermal volts/Debye length.

1. OPTIONS for printing (response 1)
   The prompt is
   ENTER SUMMARY,NX,NY
   SUMMARY -- for a summary of the number of mesh points and triangles created, how much memory allocated, and how much time used, enter 1. If not desired, enter 0.
   NX NY -- how many points along X and Y are to be printed in the print matrix. If NX and NY are both negative, a scaled integer printer plot of the potential will result. If NX and NY are both positive, a rectangular grid is placed on the region; at each point X, Y, U (the potential), UX, and UY (the derivatives of U in the X and Y directions) will be printed. At grid points outside the region, large numbers will be printed.

2. OPTIONS for plotting (response 2)

Seeing pictures is necessary if the user is (1) to understand what is happening during the iterative process leading to a solution and (2) to understand the properties of the solution.

B2DE gives the user ample opportunity to make plots. There are standard plots available of several kinds. The variables plotted are either a PDE

variable or the magnitude of its gradient. Since the author cannot guess everything the user might want to plot, B2DE has provision for user to define problem-dependent plot variables. Plots may be made to the user's graphics terminal, or the vectors may be written to a file for later processing.

Most of the graphics program code is independent of the computer used, of the graphics output device. These two dependencies are treated in the installation part of the manual.

Five kinds of plots are available: triangles, contours, surfaces, profiles, and flow lines. Each plot type has an interactive driver which is called directly from a user's program. There is an overall interactive plot driver.

All plots are only of vectors, with no plot labeling. If the hardware used supports it, erasing between plots is optional. If the hardware used supports it, the line type or color of the plots and plot background can be changed.

All the plot programs are menu-driven and use a free-format input package. As in the rest of the interactive drivers, responses of zero return the user to the next higher-up menu. Erroneous responses are dealt with reasonably; error messages should be self-explanatory. The top-level menu is:

```
1=triangles, 2=contours, 3=surface, 4=profiles, 5=flow lines, 6=erase, 7=color
```

The response should be a non-negative integer. A response of 0 means to go to the next higher menu. Other responses outside the indicated range are ignored. The responses will be treated in order.

```
        1=triangles.
```

The next menu is :

```
level, magnification (read cursor)
```

Any existing level of the triangulation can be plotted. The magnification can be unity or greater (less than unity is treated as unity); if the magnification is greater than unity, the position of the cursor is read. The cursor-reading program, TRMCRS, is terminal-dependent.

After the plot is done, the program waits for the user's signal. The signal

is terminal dependent, but is defaulted to just a carriage return. After the signal, the time taken is displayed, and the

level, magnification (read cursor)

menu comes up again. To get back to the previous menu, enter

0 0

1=triangles, 2=contours, 3=surface, 4=profiles, 5=flow lines, 6=erase, 7=color

2=contours

The next menu is :

pde no., 1-function/2-abs(grad)

Enter the pde number, from 1 through 3 the number of pdes in the problem. Enter 1 for contours of the pde value, 2 for the absolute value of its gradient. The next menu is :

number of contours, magnification (read cursor)

The number of contours is limited to 101. The magnification is treated as for triangle plots.

If nc, the number of contours, is positive, the program finds the maximum and minimum values of the plot variable and spaces the contours equally between, but not including, the extreme values. Both the extreme values and the contour values are displayed.

If nc is zero, the extreme values are displayed, but no plot is done.

If nc is negative, the user is prompted for extreme values for contours. There are -nc contours, equally spaced between and including the extreme values.

After the plot is done, the time taken is displayed, and the

pde no., 1-function/2-abs(grad)

menu comes up again. To get back to the previous menu, enter

0 0

1=triangles, 2=contours, 3=surface, 4=profiles, 5=flow lines, 6=erase, 7=color

3=surface

The next menu is :

pde no., 1-function/2-abs(grad)

Enter the pde number, from 1 through 3. Enter 1 for a plot of the pde value, 2 for a plot of the absolute value of its gradient. The next menu is :

level, 1=surface triangles, 2=surface contours, 3=surface grids

Enter the triangulation level desired for the plot, and the type of surface plot desired. In all three types, the function is plotted as a parallel projection plot (no perspective); the height of the surface above or below the x-y plane represents the function value. The surface can be viewed from any angle. Hidden lines are omitted unless the level is negative; then hidden lines are plotted; the absolute value of the level is used.

The three types will be discussed in order.

1=surface triangles

The surface is defined by the triangles of the specified level. The next menu is :

enter viewing direction -- x, y, z

The plot is displayed as if seen by a viewer looking along a line from point (x, y, z) towards the origin. After the plot is done, the time taken is displayed, and the

pde no., 1-function/2-abs(grad)

menu comes up again. To get back to the previous menu, enter

0 0

2=surface contours

The surface is defined by the contour lines drawn on triangles of the specified level. Triangle boundaries are not shown. The first menu is again :

pde no., 1-function/2-abs(grad)

and the responses are as before. The next menu is :

number of contours

Responses to this menu are like those for the planar contour plot, except no magnification is allowed. The number of contours is limited to 202.

The next menu is :

enter viewing direction -- x, y, z

Responses are as before. After the plot is done, the time taken is displayed, and the

pde no., 1-function/2-abs(grad)

menu comes up again. To get back to the previous menu, enter

0 0

3=surface grids

The surface is defined by grid lines parallel to the x and y axes, drawn on triangles of the specified level. Triangle boundaries are not shown. The first menu is again

pde no., 1-function/2-abs(grad)

and the responses are as before. The next menu is

number of x lines, number of y lines

The x lines are equally spaced along the x-axis, from the minimum x-value to the maximum x-value; the y lines are similarly spaced along the y-axis. The total number of grid lines, x plus y, is limited to 202. The next menu is :

enter viewing direction -- x, y, z

Responses are as before. After the plot is done, the time taken is displayed, and the

pde no., 1-function/2-abs(grad)

menu comes up again. To get back to the previous menu, enter

0 0

1=triangles, 2=contours, 3=surface, 4=profiles, 5=flow lines, 6=erase, 7=color

4=profiles

The next menu is :

enter pde no., number of profiles

Enter the pde number, from 1 through 3, and the number of profiles desired. A profile is a slice through a surface plot, viewed from the side. The slices are parallel and equally spaced. The next menu is :

enter indices of profile plane(s) x, y

The slices are parallel to a line running from the origin to the point (x, y). Viewing is from the direction as defined for surface plots, (y, -x, 0). After the plot is done, the time taken is displayed, and the

enter pde no., number of profiles

menu comes up again. To get back to the previous menu, enter

0 0

1=triangles, 2=contours, 3=surface, 4=profiles, 5=flow lines, 6=erase, 7=color

5=flow lines

Flow line plots attempt to describe the flux, or vector field, a sub n grad u sub n. The user specifies a line, or broken line; the positive and negative fluxes across the line are calculated; points along the broken line are spaced evenly in the flux; and lines are drawn both ways from these points. Flux lines are terminated at boundaries of the region or when the flux lines would re-enter a

triangle.

This plot does not work as well as it might. Since the solutions are composed of linear finite elements on triangles, grad u sub n is constant within each triangle. Gradients tend not to go smoothly to zero when they should, but overshoot instead; this makes the flow lines stop.

The next menu is :

enter pde no., magnification, number of points, number of flow lines

The pde number can be from 1 to npde. The magnification is like that for triangle plots and contour plots. The number of points is the number in the broken line, two or more. The next menu is :

enter N x-values followed by N y-values

where N is the number of flow lines just entered. The total flow in a positive and in a negative direction across the broken line is printed, and then the flow lines are drawn. After the plot is done, the time taken is displayed, and the

enter pde no., magnification, number of points, number of flow lines

menu comes up again. To get back to the previous menu, enter

0 0 0 0

1=triangles, 2=contours, 3=surface, 4=profiles, 5=flow lines, 6=erase, 7=color

6=erase

The DRVERA driver is called. Some terminals can support optional erasing between plots. You might want to plot, for instance, surface contours on the same plot as a surface grid, or planar contours of more than one pde on the same grid. The menu is :

1=set erase toggle off, 2=set erase toggle on, 3=erase screen

There are no lower menus. After the desired option is chosen, the main plot driver menu appears.

1=triangles, 2=contours, 3=surface, 4=profiles, 5=flow lines, 6=erase, 7=color

7=color

Some terminals can support vectors drawn in different colors or line types. The menu is :

enter new color number

There are no lower menus. After the desired option is chosen, the main plot driver menu appears.

PLOTTING TO A FILE

If the response to the menu,

1=triangles, 2=contours, 3=surface, 4=profiles, 5=flow lines, 6=erase, 7=color

is more than 100, plot vectors are sent to a file instead of the terminal. The plot type is the response modulo 100.

Users may write their own programs to read the plot file and produce plots on other devices. The format of the file is as follows. The first line is produced by the terminal-dependent program TRMPLT, and may be changed by the user. It contains the minimum and maximum integer values produced by the plot,

```
  xmin  ymin  xmax  ymax
```

in (4I6) format. These values come from the terminal-dependent program TRMSCL, and may be changed by the user; the default values are 500, 9000, 500, 7250.

The plot vectors are next in the file. Each is on a separate line,

```
    x1    y1    x2    y2
```

in (4I6) format. The end of the plot is marked by a plot vector

```
    -1    -1    -1    -1
```

which is produced by TRMPLT, and which may be changed by the user.

2. BATCH EXECUTION (for VAX-VMS only)

A. Submit File

Create a submit file which contains:

```
$ RUN MOS1.EXE
(Any input used for the the equivalent interactive
   execution.)
```

B. Output will be printed on the line printer.

3. SUMMARY OF MODEL

The MOS1 program is a portable FORTRAN 77 program suitable for analysis of currents and fields in VLSI devices. The program solves three coupled nonlinear elliptic partial differential semiconductor device equations in two dimensions. This model is implemented using adaptive finite element methods allowing a high-accuracy transistor simulation of submicron transistors.

Appendix B.

File MOS1.EXP

This example uses three multigrid levels of mesh and has a total runtime of 7185 seconds. This is a high accuracy run; running routine calculations with one or two levels of mesh will usually reduce runtime by a factor of two and still provide adequate accuracy.

```
1.E20  1          Source-Drain doping (/cm**3)  Doping flag
0.875  0.710  2.70  Vert. junc. depth, Horz. junc. depth, poly width (mu-m)
2.00  2.000 0.000  Vs Vg  Vsub (V) Id (A)  9.74604E-05 7.63260E-05
0.0556 3.9    0.0   Tox (mu-m)   Kox   Qtotal (/cm**2)
1125.0 17.0 -0.416  Mobility (cm**2/V-sec)  Channel width (mu-m)  Vfb(V)
          normalized                              unnormalized
   vd        vg        vsub              vd        vg        vsub
 7.74E+01  7.74E+01  0.00E+00          2.00E+00  2.00E+00  0.00E+00
  nch       nsd                       nch       nsd
 9.64E+05  6.66E+09                    1.45E+16  1.00E+20
  ljun      djun                      ljun      djun
 2.13E-02  2.62E-02                    7.10E-05  8.75E-05
   x1        y1        x2        y2        x1        y1        x2        y2
                                source
 0.00E+00  0.00E+00  2.13E-02  0.00E+00  0.00E+00  0.00E+00  7.10E-05  0.00E+00
                                  gate
 2.13E-02  0.00E+00  1.02E-01  0.00E+00  7.10E-05  0.00E+00  3.41E-04  0.00E+00
                                 drain
 1.02E-01  0.00E+00  1.23E-01  0.00E+00  3.41E-04  0.00E+00  4.12E-04  0.00E+00
                               substrate
 0.00E+00 -5.16E-02  1.23E-01 -5.16E-02  0.00E+00 -1.72E-04  4.12E-04 -1.72E-04
1-print, 2-plot, 3-solve, 4-comp, 5-save, 6-end
3
Number of levels
3
level  1
 Iteration  1   Normalized Current  2.87189E+09   Current (A)  3.03271E-01
 Iteration  2   Normalized Current  2.95012E+08   Current (A)  3.11532E-02
 Iteration  3   Normalized Current  6.19521E+07   Current (A)  6.54213E-03
 Iteration  4   Normalized Current  2.14547E+07   Current (A)  2.26561E-03
 Iteration  5   Normalized Current  1.53287E+07   Current (A)  1.61871E-03
 Iteration  6   Normalized Current  8.63969E+06   Current (A)  9.12349E-04
 Iteration  7   Normalized Current  5.64527E+06   Current (A)  5.96140E-04
 Iteration  8   Normalized Current  3.59749E+06   Current (A)  3.79895E-04
 Iteration  9   Normalized Current  2.02483E+06   Current (A)  2.13822E-04
 Iteration 10   Normalized Current  8.60835E+05   Current (A)  9.09040E-05
 Iteration 11   Normalized Current  5.64297E+05   Current (A)  5.95896E-05
```

Iteration 12	Normalized Current	4.91387E+05	Current (A)	5.18904E-05
Iteration 13	Normalized Current	4.84305E+05	Current (A)	5.11425E-05
Iteration 14	Normalized Current	2.24446E+05	Current (A)	2.37015E-05
Iteration 15	Normalized Current	1.70978E+05	Current (A)	1.80553E-05
Iteration 16	Normalized Current	1.36758E+05	Current (A)	1.44416E-05
Iteration 17	Normalized Current	1.64846E+05	Current (A)	1.74077E-05
Iteration 18	Normalized Current	1.68967E+05	Current (A)	1.78429E-05
Iteration 19	Normalized Current	1.68245E+05	Current (A)	1.77666E-05
Iteration 20	Normalized Current	1.69602E+05	Current (A)	1.79100E-05
Iteration 21	Normalized Current	1.71458E+05	Current (A)	1.81059E-05
Iteration 22	Normalized Current	1.67814E+05	Current (A)	1.77211E-05
Iteration 23	Normalized Current	1.61003E+05	Current (A)	1.70019E-05
Iteration 24	Normalized Current	1.56595E+05	Current (A)	1.65364E-05
Iteration 25	Normalized Current	1.55246E+05	Current (A)	1.63940E-05
Iteration 26	Normalized Current	9.83246E+05	Current (A)	1.03831E-04
Iteration 27	Normalized Current	1.67865E+06	Current (A)	1.77265E-04
Iteration 28	Normalized Current	1.74617E+06	Current (A)	1.84395E-04
Iteration 29	Normalized Current	1.61981E+06	Current (A)	1.71052E-04
Iteration 30	Normalized Current	1.62865E+06	Current (A)	1.71985E-04
Iteration 31	Normalized Current	1.62684E+06	Current (A)	1.71795E-04
Iteration 32	Normalized Current	1.58697E+06	Current (A)	1.67584E-04
Iteration 33	Normalized Current	1.58962E+06	Current (A)	1.67864E-04
Iteration 34	Normalized Current	1.59143E+06	Current (A)	1.68055E-04
Iteration 35	Normalized Current	1.59150E+06	Current (A)	1.68062E-04
Iteration 36	Normalized Current	1.59105E+06	Current (A)	1.68015E-04
Iteration 37	Normalized Current	1.59120E+06	Current (A)	1.68031E-04
Iteration 38	Normalized Current	1.59114E+06	Current (A)	1.68024E-04
Iteration 39	Normalized Current	1.59116E+06	Current (A)	1.68026E-04
Iteration 40	Normalized Current	1.59113E+06	Current (A)	1.68023E-04
Iteration 41	Normalized Current	1.59114E+06	Current (A)	1.68024E-04
Iteration 42	Normalized Current	1.59092E+06	Current (A)	1.68001E-04
Iteration 43	Normalized Current	1.59091E+06	Current (A)	1.67999E-04
Iteration 44	Normalized Current	1.59111E+06	Current (A)	1.68021E-04
Iteration 45	Normalized Current	1.59125E+06	Current (A)	1.68036E-04
Iteration 46	Normalized Current	1.59101E+06	Current (A)	1.68011E-04
Iteration 47	Normalized Current	1.59117E+06	Current (A)	1.68028E-04
Iteration 48	Normalized Current	1.59117E+06	Current (A)	1.68027E-04
Iteration 49	Normalized Current	1.59115E+06	Current (A)	1.68025E-04
Iteration 50	Normalized Current	1.59111E+06	Current (A)	1.68021E-04
Iteration 51	Normalized Current	1.59107E+06	Current (A)	1.68017E-04
Iteration 52	Normalized Current	1.59107E+06	Current (A)	1.68017E-04
Iteration 53	Normalized Current	1.59104E+06	Current (A)	1.68013E-04
Iteration 54	Normalized Current	1.59120E+06	Current (A)	1.68030E-04
Iteration 55	Normalized Current	1.59122E+06	Current (A)	1.68032E-04

```
 Iteration 56  Normalized Current  1.59091E+06   Current (A)  1.67999E-04
level  2
 Iteration  1  Normalized Current  2.13801E+06   Current (A)  2.25774E-04
 Iteration  2  Normalized Current  2.05828E+06   Current (A)  2.17354E-04
 Iteration  3  Normalized Current  2.01392E+06   Current (A)  2.12669E-04
 Iteration  4  Normalized Current  2.00789E+06   Current (A)  2.12033E-04
 Iteration  5  Normalized Current  2.00599E+06   Current (A)  2.11832E-04
 Iteration  6  Normalized Current  2.00437E+06   Current (A)  2.11661E-04
 Iteration  7  Normalized Current  2.08952E+06   Current (A)  2.20653E-04
mg  2  1.794E-02  1.794E-02  1.247E+00
mg - diverging. exit (This indicates that convergence cannot be improved)
mg  2  2.627E+02  2.627E+02  2.347E+02
mg - diverging. exit (This indicates that convergence cannot be improved)
 Iteration  8  Normalized Current  2.01056E+06   Current (A)  2.12315E-04
 Iteration  9  Normalized Current  2.00851E+06   Current (A)  2.12098E-04
level  3
 Iteration  1  Normalized Current  1.97135E+06   Current (A)  2.08174E-04
 Iteration  2  Normalized Current  1.96049E+06   Current (A)  2.07028E-04
 Iteration  3  Normalized Current  1.94898E+06   Current (A)  2.05812E-04
mg  3  1.521E+02  1.521E+02  1.164E+03
mg - diverging. exit (This indicates that convergence cannot be improved)
mg  3  4.517E+02  4.517E+02  5.766E+01
mg - diverging. exit (This indicates that convergence cannot be improved)
1-print, 2-plot, 3-solve, 4-comp, 5-save, 6-end
5
writing solution to file 23
1-print, 2-plot, 3-solve, 4-comp, 5-save, 6-end
6
```

HISTOGRAM

NUMBER OF OBSERVATIONS = 81
MINIMUM OBSERVATION = 0.170596497E-03
MAXIMUM OBSERVATION = 0.243801580E-03
HISTOGRAM LOWER BOUND = 0.165000005E-03
HISTOGRAM UPPER BOUND = 0.245000003E-03
OBSERVATIONS USED = 81
MIN. OBSERVATION USED = 0.170596497E-03
MAX. OBSERVATION USED = 0.243801580E-03
MEAN VALUE = 0.215642198E-03
MEDIAN VALUE = 0.220421032E-03
25 PCT TRIMMED MEAN = 0.220370712E-03
STANDARD DEVIATION = 0.180789702E-04
MEAN DEV./STD. DEV. = 0.758268118E+00
B1**0.5 = -.114878392E+01
B2 = 0.337107062E+01

INTERVAL MID POINT	CUM. FRACT.	1-CUM. FRACT.	CELL FRACT.	NO. OBS.	NUMBER OF OBSERVATIONS 0 10 20 30
0.170000E-03	0.049	1.000	0.049	4	++++
0.180000E-03	0.123	0.951	0.074	6	++++++
0.190000E-03	0.160	0.877	0.037	3	+++
0.200000E-03	0.185	0.840	0.025	2	++
0.210000E-03	0.346	0.815	0.160	13	+++++++++++++
0.220000E-03	0.642	0.654	0.296	24	++++++++++++++++++++++++
0.230000E-03	0.938	0.358	0.296	24	++++++++++++++++++++++++
0.240000E-03	1.000	0.062	0.062	5	+++++

Vg = 2.000 Vd = 2.000 Vsub = 0.000 Vfb = -0.416 current 2.0581E-04
Enter 0-new solution, 1-old solution, 2-quit
2
CPU time(sec.) 7185.70

NBS-114A (REV. 2-8C)

U.S. DEPT. OF COMM. BIBLIOGRAPHIC DATA SHEET (See instructions)	1. PUBLICATION OR REPORT NO. NBS/SP-400/77	2. Performing Organ. Report No.	3. Publication Date April 1985

4. TITLE AND SUBTITLE

Semiconductor Measurement Technology:
MOS1: A Program for Two-Dimensional Analysis of Si MOSFETs

5. AUTHOR(S)

Charles L. Wilson and James L. Blue

6. PERFORMING ORGANIZATION (If joint or other than NBS, see instructions)

NATIONAL BUREAU OF STANDARDS
DEPARTMENT OF COMMERCE
GAITHERSBURG, MD 20899

7. Contract/Grant No.

8. Type of Report & Period Covered

Final

9. SPONSORING ORGANIZATION NAME AND COMPLETE ADDRESS (Street, City, State, ZIP)

Same as in item 6 above.

10. SUPPLEMENTARY NOTES

Library of Congress Catalog Card Number: 85-600520

☐ Document describes a computer program; SF-185, FIPS Software Summary, is attached.

11. ABSTRACT (A 200-word or less factual summary of most significant information. If document includes a significant bibliography or literature survey, mention it here)

The MOS1 program is a portable FORTRAN 77 program suitable for analysis of currents and fields in VLSI devices. The program solves three coupled nonlinear elliptic partial differential semiconductor device equations in two dimensions. Historically, these equations have been solved using a special-purpose program and batch runs on a large, fast computer. We use a general-purpose program which runs on a large minicomputer or scientific workstation. This report discusses the physical formulation of the semiconductor equations and the methods used to select the solution strategy.

12. KEY WORDS (Six to twelve entries; alphabetical order; capitalize only proper names; and separate key words by semicolons)

finite elements; MOS transistor; MOSFET models; partial differential equations; semiconductor device model; simulation.

13. AVAILABILITY

☒ Unlimited

☐ For Official Distribution. Do Not Release to NTIS

☒ Order From Superintendent of Documents, U.S. Government Printing Office, Washington, D.C. 20402.

☐ Order From National Technical Information Service (NTIS), Springfield, VA. 22161

14. NO. OF PRINTED PAGES

61

15. Price

Announcement of Semiconductor Measurement Technology
List of Publications 72 - 1962-1984

Semiconductor Materials and Processes Division
National Bureau of Standards
Bldg. 225, Room A305
Gaithersburg, MD 20899

Please send a copy of your latest "Semiconductor Measurement Technology, List of Publications 72."

Name _____

Company _____

Address _____

City _____ State _____ Zip Code _____

NBS Technical Publications

Periodical

Journal of Research—The Journal of Research of the National Bureau of Standards reports NBS research and development in those disciplines of the physical and engineering sciences in which the Bureau is active. These include physics, chemistry, engineering, mathematics, and computer sciences. Papers cover a broad range of subjects, with major emphasis on measurement methodology and the basic technology underlying standardization. Also included from time to time are survey articles on topics closely related to the Bureau's technical and scientific programs. As a special service to subscribers each issue contains complete citations to all recent Bureau publications in both NBS and non-NBS media. Issued six times a year.

Nonperiodicals

Monographs—Major contributions to the technical literature on various subjects related to the Bureau's scientific and technical activities.

Handbooks—Recommended codes of engineering and industrial practice (including safety codes) developed in cooperation with interested industries, professional organizations, and regulatory bodies.

Special Publications—Include proceedings of conferences sponsored by NBS, NBS annual reports, and other special publications appropriate to this grouping such as wall charts, pocket cards, and bibliographies.

Applied Mathematics Series—Mathematical tables, manuals, and studies of special interest to physicists, engineers, chemists, biologists, mathematicians, computer programmers, and others engaged in scientific and technical work.

National Standard Reference Data Series—Provides quantitative data on the physical and chemical properties of materials, compiled from the world's literature and critically evaluated. Developed under a worldwide program coordinated by NBS under the authority of the National Standard Data Act (Public Law 90-396). NOTE: The Journal of Physical and Chemical Reference Data (JPCRD) is published quarterly for NBS by the American Chemical Society (ACS) and the American Institute of Physics (AIP). Subscriptions, reprints, and supplements are available from ACS, 1155 Sixteenth St., NW, Washington, DC 20056.

Building Science Series—Disseminates technical information developed at the Bureau on building materials, components, systems, and whole structures. The series presents research results, test methods, and performance criteria related to the structural and environmental functions and the durability and safety characteristics of building elements and systems.

Technical Notes—Studies or reports which are complete in themselves but restrictive in their treatment of a subject. Analogous to monographs but not so comprehensive in scope or definitive in treatment of the subject area. Often serve as a vehicle for final reports of work performed at NBS under the sponsorship of other government agencies.

Voluntary Product Standards—Developed under procedures published by the Department of Commerce in Part 10, Title 15, of the Code of Federal Regulations. The standards establish nationally recognized requirements for products, and provide all concerned interests with a basis for common understanding of the characteristics of the products. NBS administers this program as a supplement to the activities of the private sector standardizing organizations.

Consumer Information Series—Practical information, based on NBS research and experience, covering areas of interest to the consumer. Easily understandable language and illustrations provide useful background knowledge for shopping in today's technological marketplace.

Order the **above** *NBS publications from: Superintendent of Documents, Government Printing Office, Washington, DC 20402.*

Order the **following** *NBS publications—FIPS and NBSIR's—from the National Technical Information Service, Springfield, VA 22161.*

Federal Information Processing Standards Publications (FIPS PUB)—Publications in this series collectively constitute the Federal Information Processing Standards Register. The Register serves as the official source of information in the Federal Government regarding standards issued by NBS pursuant to the Federal Property and Administrative Services Act of 1949 as amended, Public Law 89-306 (79 Stat. 1127), and as implemented by Executive Order 11717 (38 FR 12315, dated May 11, 1973) and Part 6 of Title 15 CFR (Code of Federal Regulations).

NBS Interagency Reports (NBSIR)—A special series of interim or final reports on work performed by NBS for outside sponsors (both government and non-government). In general, initial distribution is handled by the sponsor; public distribution is by the National Technical Information Service, Springfield, VA 22161, in paper copy or microfiche form.

U.S. Department of Commerce
National Bureau of Standards
Gaithersburg, MD 20899

Official Business
Penalty for Private Use $300